Relatório de Segurança para Comparação de Funções – Função de Servidor de Banco de Dados

Herbert H. Thompson, Ph.D.
Fabien Casteran, M.Sc.

Junho de 2005

1990 W. New Haven Ave., Melbourne, FL 32904
Tel: (321) 308-0557 Fax (321) 308-0552
info@securityinnovation.com
www.securityinnovation.com

Sumário Executivo

Ao levar em conta a plataforma a ser implantada em uma empresa ou para designar uma função específica, os tomadores de decisões da área de TI sempre analisam um pequeno conjunto de critérios, como o preço de compra, a compatibilidade com as aplicações/tecnologias existentes, o custo de implantação e manutenção. Historicamente, a segurança sempre esteve ausente desta lista. Em muitos casos, no entanto, o custo para as empresas que têm pouca segurança e decisões de implantação gerou outros custos tradicionalmente avaliados.

Tendo isso em mente, o mercado precisa urgentemente de medidas objetivas quanto à segurança da plataforma, que são significativas em um contexto de implantação. Qualquer medida de segurança significativa *deve* ser baseada em uma visão holística de um sistema, considerando também a função que o sistema irá servir.

No início de 2004, pensamos em conduzir comparações importantes sobre a segurança quanto às soluções de plataformas em uma variedade de funções dos servidores. Nossa primeira comparação levou em conta a função do servidor Web, em que as organizações centralizam a implantação de aplicações dentro de uma única função de servidor que fornece o banco de dados, os serviços da Web e a funcionalidade de escrita dos aplicativos. No mais recente relatório, consideramos a função mais granular de um servidor de banco de dados, em que uma máquina deve apenas gerenciar, armazenar e recuperar dados de forma altamente disponível. Especificamente, comparamos três soluções distintas para preencher essas funções: o Microsoft Windows Server 2003 que executa o servidor de banco de dados Microsoft SQL Server 2000 Service Pack 3, o Red Hat Enterprise Linux 3.0 (RHEL 3.0) que executa o servidor de banco de dados MySQL[1] e o Red Hat Enterprise Linux 3.0 (RHEL 3.0) que executa o servidor de banco de dados Oracle 10g.

As comparações mais acirradas dos dados são feitas em vulnerabilidades reportadas que afetam esses sistemas, assim como os diagnósticos relevantes à segurança do sistema em geral. Mais especificamente, iremos levar em conta as vulnerabilidades desses sistemas, obtidas no período de um ano, a começar em 1.° de Março de 2004, indo até 28 de Fevereiro de 2005.

Em nossa análise, levantamos a modularidade inerente do Linux para considerar um sistema de “instalação mínima” do servidor de banco de dados do MySQL, que apresenta uma superfície menor de ataque. Para o caso do Oracle, seguimos a configuração recomendada pela Oracle de projetar o nosso servidor e considerar apenas as vulnerabilidades que afetam a configuração depois de implantada. Para a solução baseada na Microsoft, existem muitos componentes que são difíceis ou impossíveis de remover completamente do sistema operacional e, portanto, consideramos uma instalação “completa” e contamos as vulnerabilidades para todas as aplicações incluídas no software do Windows Server.

Muitos irão se lembrar do worm Slammer, de Janeiro de 2003, que atingiu o Microsoft SQL Server 2000 SP2 e outros.

[1] Servidores do Red Hat Enterprise Linux 3 vendidos com o MySQL 3.23, e esta é a versão sob manutenção corporativa da Red Hat, portanto esta é a versão analisada no relatório. A utilização de outras versões significaria que os clientes devem fornecer auto-manutenção ao servidor.

Ameaças como o Slammer foram a maior preocupação dos departamentos de TI para obter a segurança do servidor e o gerenciamento de atualizações.
Neste relatório, analisamos as principais métricas que auxiliam os administradores a examinar como a segurança operacional dos três diferentes sistemas de banco de dados surgiu no último ano.
Olhando-se apenas para as aplicações de bancos de dados, nosso estudo encontrou que o SQL Server 2000 tinha vulnerabilidade zero, naquele período de um ano, o MySQL teve 7 vulnerabilidades e o Oracle 10g tinha mais de 30.

Ao examinar a pilha completa de servidor, necessária para preencher uma função de banco de dados, nosso estudo encontrou um total de 63 vulnerabilidades para a solução baseada em SQL Server 2000, no Windows Server 2003, comparando-se com as 116 vulnerabilidades para a configuração mínima do MySQL no Red Hat Enterprise Linux, e 207 vulnerabilidades para a solução do Red Hat Enterprise Linux executada no Oracle.

Além disso, ao examinar os "dias de risco" – tempo intermediário em que uma vulnerabilidade é publicamente desvendada até quando um patch é liberado pelo fornecedor da vulnerabilidade – encontramos uma média de 32 dias de risco por vulnerabilidade da solução do Windows Server 2003 com o SQL Server 2000, 61.6 dias por vulnerabilidade da solução Linux com o MySQL e 38.7 dias de risco para a solução Linux executando o Oracle.

Esperamos que os resultados deste estudo forneçam instruções valiosas para o gerente de TI, que deve fazer a aquisição da plataforma e as decisões de implantação para maximizar o valor e minimizar o risco de segurança.

Escopo da Análise

Para obter uma visão completa dos Riscos de Segurança, uma pessoa deve levar em conta dois importantes fatores:

Vulnerabilidade do software, sistemas ou redes (onde for apropriado).

Ameaças contra essas vulnerabilidades

Desses dois fatores, nossa experiência própria nos leva a crer que o último é mais difícil de se quantificar ou prever de maneira objetiva. Este é um campo interessante e aberto, e incentivamos outras pessoas a levar em conta essa área, a fim de obter uma pesquisa mais reflexiva. No entanto, uma vez que há oportunidades de pesquisas nas duas áreas, optamos por tentar e progredir nos estudos e medidas dos fatores de vulnerabilidades primeiro; essa é uma métrica precursora para outras que se baseiam nas ameaças. Dessa forma, não levamos em consideração o perfil de ameaça nesse estudo e, em vez disso, irmos nos concentrar na vulnerabilidade subjacente do sistema.
Conforme foi observado no nosso estudo anterior, Web Study,[2] vamos nos voltar às medidas comparativas focadas no cliente quanto à segurança das plataformas – em particular, analisamos essas medidas, as quais os fornecedores têm a capacidade de afetar e aprimorar. Este estudo atual enfatiza o objetivo e concentra-se em fornecer as principais visões para a segurança das plataformas mais comuns em uma função do servidor de banco de dados.

[2] O relatório de Análise da Security Innovation "Relatório da Comparação entre Funções – Função do Web Server," Março de 2005, http://www.sisecure.com/pdf/windows_linux_final_study.pdf

Agradecimentos

Este estudo e as nossas análises foram baseados em um contrato de pesquisa a partir da Microsoft. Como parte do contrato, temos controle editorial total sobre as pesquisas e análises aqui apresentadas. Estamos atrás da nossa metodologia e execução quanto àquela metodologia de se determinarem resultados objetivos aos clientes e praticantes da segurança.

Incentivamos outras pessoas a examinar, analisar bastante e comentar nosso trabalho. Nosso objetivo foi de desempenhar uma análise baseada em fatos (e não em especulações) usando uma metodologia transparente e significativa. A síntese de feedback no surgimento da nossa pesquisa beneficia todas as pessoas, independente da preferência: nosso objetivo, nesta abordagem, é de trazer esclarecimentos, e não polêmicas.

Sendo assim, agradecemos Mark Cox, da Red Hat, por ter trabalhado conosco para ajudar a resolver as discrepâncias entre as datas de suas próprias vulnerabilidades e outras que compilamos usando a metodologia publicada em Março de 2005. O feedback de Mark foi extremamente valioso e nos ajudou a refinar nossa metodologia (veja o Apêndice A) para acomodar os bancos de dados privados, trazendo, posteriormente, conjuntos mutuamente valiosos de dados, usados em todas as análises dentro deste estudo.

Este estudo é a segunda parte das nossas tentativas comparativas da segurança de plataforma, sob uma variedade de funções. Ao lançar nosso primeiro estudo nas comparações de funções sobre Web Server, em Março de 2005, recebemos feedbacks positivos das comunidades acadêmicas, usuários e analistas, e gostaríamos, portanto, de agradecer todos os que nos forneceram esses feedbacks. Esses comentários foram incorporados ao longo deste documento e esperamos que este relatório também seja comentado.

Por fim, queremos agradecer Richard Ford, do Instituto de Tecnologia da Flórida, pela co-criação da metodologia subjacente aqui utilizada, assim como sua contribuição extensiva aos pontos analíticos desses estudos.

Introdução

A segurança é uma séria preocupação para aqueles que implantam sistemas modernos de computadores – isso é verdade, pois a segurança relativa das diferentes soluções pode ser fator principal ao se escolherem plataformas e aplicações. Além disso, tendo-se certos ciclos de vida da implantação de um produto, muitas empresas fazem escolhas que irão afetar suas operações para os próximos dez ou quinze anos.

Neste documento, apresentamos uma comparação baseada em funções da segurança relativa de três soluções diferentes de uma função do Servidor de Banco de Dados. Ao se concentrar nessas *funções,* é possível criar comparações de servidor que sejam significativas e centradas nos requisitos dos clientes. Acreditamos que este principal aspecto tenha estado ausente das comparações quantitativas anteriores; sua inclusão fornece mais comparações significativas das funcionalidades equivalentes.

O gerenciamento dinâmico, o armazenamento e a recuperação de dados são algumas das funções mais críticas de servidor em diversos ambientes. Essa função existe a partir de uma série ampla de indústrias e neste documento, então, presumimos que uma organização tenha um objetivo de alto nível ao possuir um sistema de banco de dados extensível, robusto, com alto desempenho e disponibilidade e que existam múltiplas plataformas/servidores que possam suprir esses requisitos.

Para esta finalidade, primeiro fornecemos uma descrição geral da função do Servidor de Banco de Dados e depois descrevemos uma implantação típica desta função, usando o Windows Server 2003 com o Microsoft SQL Server 2000, Red Hat Enterprise Linux (RHEL) 3 com o servidor de banco de dados do MySQL e o Red Hat Enterprise Linux 3 com o Oracle 10g. Em um nível mais alto, descrevemos a nossa abordagem dos requisitos de tais funções e, posteriormente, a usamos como um roteiro para implementações específicas necessárias para as soluções do Windows Server 2003 e RHEL. As comparações da segurança – tanto quantitativas como qualitativas – são feitas a partir desta função nas configurações.

Por fim, iremos concluir, checando os resultados e as ações que os fornecedores devem tomar para aprimorar sua segurança, se medida por esta metodologia.

Descrição/Definição da Função do Servidor de Banco de Dados

A necessidade de um gerenciamento de dados altamente disponível e confiável é muito importante para as organizações de hoje em dia. Os servidores de banco de dados suprem essa necessidade, fornecendo às aplicações o acesso às informações de uma forma organizada e útil. Para suprir as demandas corporativas, tais servidores devem encontrar um grande espectro de requisitos de usuário. Tipicamente, os servidores de bancos de dados interagem tanto com a internet e a intranet, lidando com os componentes e, portanto, a necessidade pela segurança e tolerância a falhas é alta. Além do sistema operacional de hospedagem, para a função do servidor de banco de dados nós precisamos de uma aplicação de servidor e um software que suportem a criptografia dos dados transmitidos.

Suporte para SSL/Criptografia

Acreditamos que, para fins de segurança e privacidade, uma função típica de servidor de banco de dados precise de uma implementação sólida de tecnologia de criptografia; em particular, um suporte deve existir para os padrões aceitos, como o SSL e o TLS para a transmissão de dados.

Isso é importante para proteger os dados sensíveis durante a transferência entre o cliente e o servidor, ou entre as camadas de aplicação.

O servidor de banco de dados

O servidor de banco de dados é responsável pelo armazenamento e recuperação dinâmica das informações. O servidor deve fornecer uma grande série de interfaces para interagir com outros servidores, como os servidores de aplicações, de arquivo e impressão e outros sistemas. Ele deve, por exemplo, ser equipado para controlar uma conexão ODBC e receber e responder às consultas do SQL.

Uma Plataforma de Servidor

Examinar um servidor de banco de dados em isolamento, a partir de um sistema operacional subjacente, seria ignorar a base que suporta a função do banco de dados. Um banco de dados seguro perfeitamente instalado em uma plataforma insegura não será capaz de manter sua segurança. Portanto, devemos levar em conta a *segurança total da solução* em nossas análises e depois incluir os problemas da plataforma. Neste estudo, nós analisamos, especificamente, o Windows Server 2003 e o Red Hat Enterprise Linux 3.

Plataformas Comparadas

Em nossa análise, escolhemos plataformas para comparar o que seria mais significativo aos clientes. Tendo-se o alto nível de interesse no Windows versus Linux, escolhemos a mais recente versão do software de servidor do Windows, o Microsoft Windows Server 2003 e o Red Hat Enterprise Linux 3[3][4]. Notamos que há muitas distribuições diferentes do Linux que poderíamos selecionar, mas os relatórios mais recentes de analistas indicam que os clientes corporativos estão optando muito mais pelo Red Hat ou SuSE nos ambientes de produção. Dessa forma, selecionamos o Red Hat para estes testes, pois ele é líder em distribuição[5]. Além disso, devido à forte posição desta solução no mercado, o Red Hat vem sendo o melhor representante dos distribuidores de código aberto e o candidato preferido em nossas análises.

Originalmente, aplicamos essa metodologia à função do Web Server, em um relatório lançado em Março de 2004, que se concentra nas funções projetadas no Windows Server 2003 e o Red Hat Enterprise Linux. O relatório atual limita-se aos resultados dessas mesmas plataformas na função do Servidor de Banco de Dados. Enquanto os dois relatórios analisam as mesmas plataformas, eles seguem uma metodologia genérica que poderia ser aplicada a outros produtos, fornecedores e funções de servidor.

[3] Embora analisemos a versão ES do Red Hat Enterprise Linux 3, os pacotes instalados na versão AS são muito semelhantes à plataforma para poder ser feita uma comparação.
[5] Fonte: relatório do IDC, “Ambientes Operacionais do Linux no Mundo Todo 2004-2008 Previsão e Análise: Produtos Empresariais de Olho no Futuro”, Dezembro de 2004.
[4] Uma vez que este estudo foi concluído, o Red Hat lançou o Red Hat Enterprise Linux 4. Confira a “Observação de Análise” abaixo para mais detalhes.

Red Hat Enterprise Linux ES 3

O Red Hat é líder mundial de soluções de código aberto empresarial[5]. Devido à sua forte posição no mercado de soluções empresariais do Linux, o Red Hat é o melhor representante dos distribuidores de código aberto e o candidato favorito para essas análises.

Entre as soluções, encontra-se o Red Hat Enterprise Linux ES 3, que é a solução empresarial que o Red Hat recomenda para as pequenas a médias configurações da web. Após um IPO de sucesso, em 1999, o Red Hat tira proveito de sua marca consolidada, sendo considerado por muitos o nome mais reconhecido nas distribuições de Sistemas Operacionais (OS) de código aberto. Além disso, o Red Hat Enterprise Linux é amplamente implantado na função do servidor da web, o que o torna um candidato óbvio e significativo para análises.

O Red Hat Enterprise Linux 3 utiliza uma abordagem híbrida de kernel com recursos do Linux 2.6 kernel *back-ported* (de porta de trás) para utilização com o kernel estável do Linux 2.4. O RHEL ES 3 inclui suporte para diversas arquiteturas e fornece recursos e suporte exigidos pelas grandes organizações.

A fim de projetar um servidor de banco de dados funcional, usando o RHEL, devemos primeiro escolher componentes diferentes que supram os requisitos funcionais nas próximas seções deste relatório. Abaixo está uma lista dos principais componentes, juntamente com uma breve justificativa de sua seleção.

OBSERVAÇÃO da Análise: O Red Hat veio com o Red Hat Enterprise Linux 4 (RHEL4), como o sucessor do RHEL3, em Janeiro 2005, mas, sob esses acordos empresariais, o RHEL3 será suportado por muitos anos. Considerando que queríamos estudar, no mínimo, 12 meses de dados, o RHEL4 não seria uma boa opção.

Devido à inclusão do kernel 2.6 SELinux no RHEL4, ocorre certa especulação de que a "segurança aprimorada"mudaria significativamente os resultados deste estudo, caso fosse possível usar o RHEL4.

A "Segurança Aprimorada" do kernel 2.6 refere-se a uma nova capacidade de controle de acesso baseado em função e não à qualidade do código, portanto, vale mais a pena adquirir certo espaço para ver como as vulnerabilidades e os diagnósticos estão agindo no RHEL 4.

De acordo com o site do Red Hat, entre 15 de Fevereiro e o final de Abril, de 2005, o Red Hat emitiu o seguinte:

36 consultas de segurança do Red Hat Enterprise Linux 3 AS, e

61 consultas de segurança do Red Hat Enterprise Linux 4 AS

Embora essas consultas representem todos os pacotes das versões do Red Hat Enterprise Linux e não apenas um conjunto mínimo de pacotes que estudamos, os dados desta estrutura de tempo (timeframe) limitada não parece indicar uma melhoria drástica na vulnerabilidade da segurança do RHEL4.

Além disso, as três vulnerabilidades expostas no MySQL durante o período após o lançamento do RHEL4 (e dentro do nosso período de estudo) afetaram o MySQL, quando ele acompanhava o RHEL3 e o RHEL4.

Red Hat Enterprise Linux 3 – Servidor de Banco de Dados do MySQL

Ao passo que existem diversas opções potenciais para um servidor de banco de dados de código aberto, como o postgreSQL e o MySQL, optamos por analisar o servidor do MySQL devido à sua popularidade no mercado. Nos resultados deste estudo, no entanto, examinamos as diferenças de vulnerabilidades entre o MySQL e o postgreSQL.

Seguem algumas aplicações requeridas para suprir a função do servidor de banco de dados na plataforma Red Hat com o MySQL:

Servidor de banco de dados: MySQL

Mais de 4 milhões de instalações ativas do MySQL no mundo tornam o servidor de banco de dados do MySQL o banco de dados de código aberto mais popular[6]. Os usuários deste servidor podem optar por utilizá-lo sob a Licença Pública Geral, GNU, ou sob uma licença comercial. O MySQL é leve e pode controlar diversas conexões de forma rápida e confiável. Ela não apresenta alguns recursos de outros servidores de banco de dados (como o PostgreSQL), como visualizações e sub-consultas, no entanto, seu desempenho é superior à maioria dos concorrentes de código aberto. O MySQL também é o componente de banco de dados da tão conhecida configuração "LAMP"[7] , um conjunto de programas gratuitos comumente usados juntos para executar web sites e ships dinâmicos como um componente suplementar do RHEL ES 3.

Red Hat Enterprise Linux 3 – Servidor de Banco de Dados da Oracle

A Oracle se estabeleceu como provedor líder em tecnologias de banco de dados no mercado de servidores da plataforma Linux[8]. Um relatório recente, pela IDC, uma empresa analista de tecnologia, estima que a Oracle retém 41.3% da participação de mercado quanto aos bancos de dados relacionais. A Oracle Corporation é o maior fornecedor de software comercial, atingindo a marca de U$10 bilhões no Ano Fiscal de 2004.

Para essa configuração, instalamos o Oracle de acordo com suas diretrizes de implantação[10] e então analisamos o sistema resultante. Seguem algumas aplicações requeridas para suprir a função do servidor de banco de dados na plataforma Red Hat com o Oracle:

Servidor de banco de dados: Oracle 10g

---

[6] Estudo da BZ Media, "Database and Data Access, Integration and Reporting Study", http://www.bzmedia.com/bzresearch/5914.htm, Maio de 2004.

[7] LAMP, ou "Linux, Apache, MySQL, Perl/PHP/Python" é considerado "padrão" na configuração do código aberto de um servidor dinâmico.

[8] Oracle corporation, http://www.oracle.com/database/feature_db_dbleadership.html

[9] IDC Corporation, "Robust Recovery in Worldwide RDBMS Market, But Results Tempered by Currency Environment, IDC Reveals," http://www.idc.com/getdoc.jsp?containerId=prUS00089505

[10] Oracle corporation, "Installing Oracle Database 10g on Linux x86," http://www.oracle.com/technology/pub/articles/smiley_10gdb_install.html

Com sua última versão 10g do produto, a Oracle fornece uma plataforma robusta e extensível, com suporte para o gerenciamento e configuração dos dados de servidor, a fim de encontrar uma série diversa de necessidades dos usuários.

Microsoft Windows Server 2003

O Windows Server 2003 é um sistema operacional de servidor que pode assumir diferentes funções, incluindo a função de servidor de banco de dados. A plataforma Windows Server 2003 é muito bem suportada por uma empresa estabelecida, que leva a consolidada marca do Microsoft Windows. Sendo assim, ele fornece estabilidade, suporte e gerenciamento para uma implantação comercial.

Para a plataforma Microsoft Windows Server 2003, sob a função de servidor de banco de dados, tem-se a seguinte configuração:

Servidor de banco de dados: SQL Server 2000 Service Pack 3

O SQL Server 2000 é o servidor de banco de dados da Microsoft. Ele inclui todos os recursos que alguém pode esperar de um servidor de banco de dados, além das funcionalidades atribuídas para aumentar a usabilidade e o desempenho. Iremos analisar o SQL Server 2000 SP3, a versão disponibilizada quando o Windows Server 2003 acompanhou o 2003.

Requisitos

Comparar a segurança de um servidor baseado em uma análise que leva em consideração todos os problemas de segurança reportados em todas as aplicações que *devem ser instaladas* neste sistema operacional não é algo nem realista e nem útil para os tomadores de decisão. Na verdade, poucos usuários possuem todos os componentes de servidor instalados ou sendo executados em seus sistemas. Assim, o foco deste relatório é comparar a segurança dos sistemas configurados na função do servidor de banco de dados, uma vez que isso é muito mais representativo para os cenários do "mundo real".

Uma das forças de segurança geralmente mencionadas quanto ao Linux é a sua modularidade, que permite uma "projeção mínima" verdadeira da função do servidor, reduzindo, portanto, sua superfície efetiva de ataque e tornando-o mais seguro. No entanto, os administradores devem operar dentro dos limites de gerenciamento e também implantar de uma forma que ainda seja suportada pelos fornecedores. Para a configuração do Oracle ou do Linux, instalamos tanto o sistema operacional (RHEL 3) como o servidor de banco de dados (Oracle 10g) de acordo com o método exato especificado pelo Oracle[11], que é detalhado no Apêndice B deste relatório. Para construção do MySQL no Linux, levantamos a modularidade do Linux para fornecer uma instalação mínima (e, portanto, uma menor superfície de ataque) para este servidor.

Na construção do SQL Server 2000 e do Microsoft Windows Server 2003, os componentes, como o Internet Explorer, estarão presentes nas cargas de trabalho, uma vez que não possam ser facilmente desinstalados da função do servidor de banco de dados. Em nossa pesquisa,

---

[11] http://www.oracle.com/technology/pub/articles/smiley_10gdb_install.html

assumimos que um cliente precisaria diagnosticar *qualquer* questão que esteja presente no software do servidor da Microsoft. Portanto, enquanto os problemas com o Internet Explorer irão contar contra a construção a Microsoft, as vulnerabilidades semelhantes que afetam o Mozilla não contarão pontos contra a construção do MySQL no RHEL 3. Em nosso estudo, implantamos o software e validamos fisicamente as configurações.

Para analisar a segurança, levamos em conta tanto as métricas quantitativas como as qualitativas. Nossa análise irá então avaliar as três soluções em ambas as frentes, assim como irá examinar, à parte, as aplicações de banco de dados.

Descrição e Introdução das Métricas Quantitativas

Historicamente, as comparações de segurança entre plataformas foram desempenhadas em dados prontamente disponíveis e quantitativos. Tais comparações foram apenas consideradas “boletins de segurança” ou “pesquisas de segurança” emitidas pelos fornecedores.

Enquanto as pesquisas são populares, poucas provas existem para suportar sua inutilidade em tomar decisões de implantação referentes à segurança, uma vez que os fornecedores controlam quantas vulnerabilidades devem ser emitidas por uma única pesquisa. Esses boletins de pesquisa, portanto, não representam a qualidade subjacente de segurança dos produtos, a menos que um cuidado extra seja tomado para assegurar que o comportamento do fornecedor seja semelhante. Por exemplo, o SuSE Enterprise Linux e o Red Hat Enterprise Linux possuem um alto nível de correlação entre os componentes. No entanto, embora ambos os ajustes sejam semelhantes, de vulnerabilidades principais dos componentes (ex. Linux kernel, X Windows, drivers de hardware, rsync), o número de pesquisas de segurança do SuSE é significativamente menor, devido a seu uso de pesquisas sintéticas. Se alguém precisasse desempenhar uma análise simples de pesquisa, iria representar dados melhores – porém enganosos – da segurança relativa do SuSE quanto ao Red Hat.
Na verdade, o número de vulnerabilidades ajustadas por cada fornecedor tem a mesma dimensão, mas o Red Hat é mais granular (e até mais transparente) na liberação das pesquisas de segurança.

Em vez disso, tomamos uma abordagem baseada em funções para medir a segurança com base em configurações provavelmente implantadas. Para os dados quantitativos, esta abordagem significa apenas considerar essas vulnerabilidades e patches que se aplicam a uma função implantada. Por exemplo, não devemos considerar uma vulnerabilidade reportada em um componente que não esteja instalado em nossa solução de servidor de banco de dados em funcionamento.

Período de Tempo

A metodologia usada para esta comparação poderia ser aplicada a qualquer período de tempo. Para a nossa comparação entre funções de servidor de banco de dados, iremos considerar apenas as vulnerabilidades para as quais o fornecedor de soluções (Microsoft, Oracle ou Red Hat) tenha liberado ajustes entre 1.° de Março de 2004 e 28 de Fevereiro de 2005, tendo um ano para implantação. Iremos incluir vulnerabilidades desvendadas antes de 1.° de Março de 2003, apenas se elas foram ajustadas durante esse período. Da mesma forma, não iremos considerar as vulnerabilidades *descobertas* dentro deste período de um ano, mas ajustadas após 28 de Fev de 2005. Alguém poderia selecionar períodos diferentes no futuro e repetir esses estudos utilizando novos períodos e começar a estudar as tendências também.

Enquanto o nosso objetivo era de aplicar métricas sobre o período de um ano, essas datas particulares foram escolhidas por representarem as informações mais atuais que pudessem ser analisadas no período do nosso estudo.

Descrição e Introdução das Métricas Qualitativas

Além das atualizações e vulnerabilidades, há qualidades mais "leves" de segurança, que são difíceis de se quantificar, mas que causam claro impacto na segurança implantada. As qualidades como o suporte do ciclo de vida da segurança, recursos padrões de segurança; todas têm impacto sobre a segurança da função implantada. Em nosso relatório, iremos descrever de que forma as duas soluções se comparam nesses critérios, para a conveniência dos leitores.

Hipóteses & Regras

Como indicado anteriormente, iremos analisar três configurações básicas de servidor de banco de dados, sendo uma no Microsoft Windows Server e duas no Red Hat Enterprise Linux.

MySQL no Red Hat Enterprise Linux ES 3 Para esta configuração, consideramos uma instalação mínima, em que o servidor é especificamente projetado para suprir a função, instalando apenas um conjunto mínimo de componentes of componentes ao se elaborar o sistema. O cenário de instalação mínima fornece uma superfície de ataque reduzida para o servidor, supondo que um administrador, que priorize a segurança acima de outros critérios, implante o servidor.

Oracle 10g no Red Hat Enterprise Linux ES 3 O Oracle fornece recomendações detalhadas de instalação para os usuários instalarem o Oracle 10g no RHEL 3. Enquanto os passos adicionais podem ser tomados para fortalecer o servidor ou remover certos componentes, seguimos os procedimentos recomendados pelo fornecedor, uma vez que este é, provavelmente, o cenário mais comum, estando dentro das especificações de suporte da Oracle.

SQL Server 2000 no Windows Server 2003 Como a construção da Microsoft não torna fácil remover componentes adicionais do sistema operacional do Servidor, a projeção e análise do nosso sistema incluem todos os componentes que acompanham o Windows Server 2003.

Além da instalação, segue uma lista de preocupações e necessidades do usuário que se interessar por esta análise:

O usuário requer os recursos, a confiança, o suporte e a manutenção profissional fornecidos por um distribuidor confiável de software. Por exemplo, no caso de uma solução de código aberto, como o Red Hat Enterprise Linux, presumimos que, no que se refere ao gerenciamento e ao seu custo associado, os usuários irão apenas instalar versões dos componentes do OS garantidos e liberados pelo fornecedor do OS, para que seu contrato de suporte permaneça válido.

Espera-se que o servidor de banco de dados acomode uma grande área de componentes, interfaces de aplicações e plataformas, e forneça dados de alto desempenho, de forma robusta e confiável.

Os diferentes componentes do servidor de banco de dados devem ser compatíveis e facilmente integrados.

0 sistema de gerenciamento do banco de dados deve ser capaz de executar consultas, simples e complexas, de forma rápida e confiável.

Ao levar em conta a segurança relativa de diferentes soluções, é importante não apenas considerar *o que* está instalado, mas também *como* está instalado. Assim, os recursos padrões de segurança – essencialmente, o *contexto* em que uma função é implantada – são importantes quando se considera a viabilidade de segurança em longo prazo de uma solução.

Entre as informações contextuais importantes com relação à segurança, estão:

Portas abertas na implantação padrão/configurada
Usuários (e seus privilégios)
Outros aplicativos que podem modificar o comportamento com relação às vulnerabilidades
Tecnologia que reduz a vulnerabilidade de um sistema (ex. proteção contra estouro de buffer)
Considerações ambientais (como em "um servidor que satisfaz essa função está geralmente localizado atrás de um firewall")
Considerações gerais da superfície de ataque, como o nível de privilégio dos serviços apresentados.

Hipóteses Adicionais sobre os Dados Quantitativos

Os dados quantitativos disponíveis estão relacionados com as vulnerabilidades, as atualizações e a qualidade de atualizações (patches). Essas informações podem ser obtidas a partir de listas de relatórios públicas sobre bugs e dos fornecedores que liberam esses patches. Enquanto esses resultados representam apenas a dimensão da vulnerabilidade do risco de segurança, eles fornecem uma idéia quanto aos aspectos da qualidade de segurança que estão sob o controle do fornecedor – a qualidade de segurança do código e a resposta de segurança. Essas métricas, no entanto, devem ser consideradas juntamente com diversos outros fatores qualitativos importantes ao se escolher uma plataforma baseada no custo da manutenção de segurança e na probabilidade das brechas na segurança.

Para se fazer uma comparação imparcial entre duas plataformas, a série de hipóteses usadas para reunir os dados é essencial. Essas informações também são fundamentais para tornar a experiência reprodutiva. Segue uma série de hipóteses que foram usadas para reunir informações de vulnerabilidades e de patch.

i. Presume-se que os clientes do Red Hat instalam apenas os patches liberados pela Red Hat ou Oracle (no caso de uma configuração em Oracle) e vêm tomando outros passos para garantir que estejam de acordo com o contrato de manutenção. Da mesma forma, os clientes do Windows apenas utilizam ajustes liberados pela Microsoft.

ii. Todos os ajustes liberados pelo Red Hat, Oracle e Microsoft, pertencentes aos sistemas, serão gravados juntamente com as aplicações às quais os patches pertencem. Para cada ajuste, as vulnerabilidades solucionadas serão inseridas com o uso do identificador de vulnerabilidades Mitre[12] (ex. CVE-2004-0079), caso haja alguma atribuição.

[12] O banco de dados de Vulnerabilidades e Exposições Comuns (CVE) é um padrão amplamente aceito para identificar vulnerabilidades específicas. O CVE está disponível online em http://www.cve.mitre.org. Veja também a seção “Lista CVE Mitre” abaixo.

iii. Para cada vulnerabilidade, consideramos sua classificação de severidade atribuída pelo sistema de classificação de vulnerabilidades, o Instituto Nacional de Padrões (NIST) ICAT. Enquanto os fornecedores possuem seus próprios sistemas de classificação para a severidade, esses esquemas não são necessariamente comparáveis e, assim, nós utilizamos essa fonte independente para facilitar as comparações significativas dos fornecedores cruzados.

iv. Quando uma aplicação é instalada por padrão, todas as versões atualizadas serão consideradas aplicações padrões também.

v. Para fins do período de tempo que estudamos, presumimos que os sistemas são completamente diagnosticados para a data do estudo. Por exemplo, ao olhar para o período de 1.° de Março de 2004 até 28 de Fevereiro de 2005, presumimos que todos os patches, da data inicial de nosso período, tenham sido implantados em ambos os sistemas.

vi. A “primeira data” ou descoberta pública é aquela em que a vulnerabilidade foi lançada pela primeira vez em uma lista ou site público (Bugtraq, Red Hat, Microsoft, Full-disclosure, k-otik, etc.) dedicado à segurança, ou a uma lista acessível pública de bugs ou problemas reportados pelo site doméstico de um pacote ou sua lista de mensagens. Não consideramos públicas as discussões sobre o “fornecedor-sec” Linux. E também não consideramos discussões sobre o bugzilla como descoberta pública.

vii. As datas dos patches são baseadas na liberação da distribuição de interesse.

viii. As datas de liberação para o patch ou ajuste da vulnerabilidade são específicas à distribuição/arquitetura. Caso o ajuste de um componente (ex. `libpng`) seja liberado em
01/01/1970 para certa distribuição do Linux (ex. Gentoo Linux) e um ajuste para o mesmo problema seja lançado para o Red Hat em 10/01/1970, a data de liberação deste ajuste no
Red Hat será 10/01/1970. Isso não se aplica à plataforma Windows.

ix. Para os problemas antigos, a data de liberação de um patch é o primeiro relatório publicado pelo fornecedor que inclua o patch para a plataforma para a qual o patch ajustou totalmente a vulnerabilidade. Se o patch precisou ser reemitido para consertar parte do problema de segurança, será usada a última data.

x. Os pacotes de documentação não serão inseridos. Isso se aplica apenas à plataforma Red Hat.

xi. 80x86 é a única arquitetura considerada para a plataforma Red Hat.

xii. Presume-se que os patches sejam instalados na ordem de liberação.

xiii. Os bugs funcionais são apenas registrados como vulnerabilidades no caso de os patches apresentarem-nos.

xiv. Se uma vulnerabilidade possuir um ID único, mas está em *pacotes diferentes* da aplicação de *diferentes patches*, tais vulnerabilidades devem ser contadas como o número de patches que se aplicam aos componentes instalados. Por exemplo, se uma vulnerabilidade tiver sido atribuída a um único CAN ID, mas tiver afetado duas aplicações separadas, *a* e *b*, e um patch tiver sido lançado separadamente para *a* e *b*, isso será contado como duas vulnerabilidades. Enquanto há apenas alguns

desses incidentes em nossa análise de dados, acreditamos que este seja o tratamento mais cauteloso, uma vez que múltiplos patches e pontos de exposição recorrem diretamente aos pontos fracos da segurança do usuário.

xv. Se o fornecedor do produto não participar ativamente do programa Mitre CVE, consideramos a enumeração do próprio fornecedor como a contagem de vulnerabilidades daquela aplicação. Especificamente, consideramos as vulnerabilidades do Oracle conforme enumeradas pelos boletins de segurança da Oracle. Uma vez que tais vulnerabilidades não terão classificação independente de severidade (ICAT ou CERT), consideramos que todos os problemas estejam com classificação de "nenhuma" na escala de severidade do ICAT.

Para compilar os dados em um local centralizado, criamos um banco de dados que foi preenchido usando fontes diferentes e consolidadas, utilizando nomes de identificação do CVE em que os identificadores de fornecedores estão disponíveis. Criando esse banco de dados, temos a capacidade de pesquisar facilmente qualquer tipo de dados, vulnerabilidades e patches baseados em uma quantidade de critérios diferentes, como a implicação da vulnerabilidade, função ou dias de risco.

Lista Mitre CVE

Em nossa análise, geralmente nos referimos ao CVE ou identificador CAN de uma vulnerabilidade. CVE significa Common Vulnerabilities and Exposures, sendo uma taxonomia que pretende padronizar a nomenclatura para todas as vulnerabilidades e exposições publicamente conhecidas.

Inicialmente, as vulnerabilidades são atribuídas a um número do candidato (CAN). Esses candidatos são então examinados pelo Conselho Editorial da CVE, composto de especialistas do mercado, em que uma decisão é tomada para inclusão no CVE. O CVE é mantido pela Mitre Corporation, uma organização sem fins lucrativos que desempenha pesquisas e análises independentes para o Governo dos Estados Unidos. Em nossa análise, referimo-nos a uma vulnerabilidade como sendo diferente caso tivesse seu próprio identificador CVE ou CAN. No caso dos fornecedores que não participam ativamente do programa CVE, somos obrigados a usar a própria enumeração dos fornecedores quanto às vulnerabilidades na segurança.

Enquanto a Microsoft e a Red Hat são participantes ativos do programa CVE, a Oracle não é. Nossa contagem para as vulnerabilidades da Oracle é baseada em seu sistema próprio de numeração, ligado a alertas específicos que eles liberam. Acreditamos que usar esses identificadores fornece-nos uma contagem apurada das vulnerabilidades, mas isso torna complexo o processo de relacionar diretamente uma descoberta pública de um não fornecedor a uma certa vulnerabilidade. Ainda utilizamos a abordagem conservadora de contar essas vulnerabilidades com 0 dia de risco, que, em últimos casos, pode favorecer a Oracle quanto aos dias de risco.

Classificações de Severidade do NIST ICAT

Um dos tópicos mais polemicamente debatidos sobre a segurança e a severidade de impacto que uma vulnerabilidade pode ter. O Instituto Nacional de Padrões introduziu o ICAT Metabase, que contém informações sobre as vulnerabilidades conhecidas e utiliza os identificadores CVE para catalogar esses registros. Um aspecto interessante do ICAT é a classificação de severidade que ele atribui às vulnerabilidades. As classificações do ICAT geralmente são tidas com uma forma aceita e objetiva para os administradores de sistema e profissionais de TI, a fim de estimar o impacto de uma vulnerabilidade sobre um sistema comum. Embora as classificações de severidade do ICAT não ofereçam instruções contextuais para a severidade particular de uma vulnerabilidade em certo contexto, elas fornecem uma forma objetiva de classificar vulnerabilidades. Dentro dessa possibilidade,

utilizamos as classificações de severidade do ICAT em nossa análise para agrupar as vulnerabilidades em classes. O ICAT utiliza três amplas classes de severidade para as vulnerabilidades: Alta, Média e Baixa, conforme definidas abaixo[13]:

> Uma vulnerabilidade é de "severidade alta" se:
>
> > Permite que um atacante remoto danifique a proteção de segurança de um sistema (i.e. consiga algum tipo de conta de usuário ou raiz), permite um ataque local que obtenha controle total sobre o sistema, é importante o suficiente para ter uma pesquisa CERT/CC associada. Uma vulnerabilidade é de "severidade média" se: Não encontra a definição tanto da severidade "alta" e nem "baixa". Uma vulnerabilidade é de "severidade baixa" se:
> > a vulnerabilidade não apresenta informações valiosas ou controle sobre o sistema, mas, em vez disso, fornece o conhecimento de que o atacante precisa para encontrar e explorar outras vulnerabilidades.
> >
> > Entendemos que a vulnerabilidade não provoca conseqüências para a maioria das organizações.

Vulnerabilidades não classificadas – Enquanto o ICAT contém classificações de severidade para a maioria das vulnerabilidades no CVE, há muitas que ainda permanecem sem classificação. As classificações são continuamente atualizadas no site e as informações de classificação deste estudo estão atualizadas desde 27 de Janeiro de 2005. Ao avaliar as estatísticas apresentadas adiante neste relatório, com relação à severidade, sugerimos que você trate as vulnerabilidades com a classificação de "Desconhecida" com extremo cuidado, pois elas têm o potencial para ser de severidade alta. No caso da Oracle, como não existe um mapeamento direto de nomes CVE às vulnerabilidades individuais,[14] e, uma vez que o ICAT indexa seu banco de dados pelos CVEs, consideramos que todas as vulnerabilidades da Oracle tenham uma classificação de "Desconhecida".

Sobre o cuidado com as classificações – Ao examinar as vulnerabilidades, há certa tendência de se ignorar ou subestimar aquelas classificadas como "Média" ou "Baixa". É errado pensar assim, uma vez que já analisamos diversos ataques que exploraram vulnerabilidades classificadas como baixa para obter controle remoto total sobre uma máquina. A combinação sinérgica das vulnerabilidades classificadas como baixas e médias pode levar a uma exposição da maior severidade. Outro problema é a severidade contextual da vulnerabilidade. Os sistemas de classificação, como o ICAT, atribuem rótulos de severidade baseados em seu impacto potencial ou aplicação separadamente. Essas classificações oferecem uma idéia mínima no impacto de *uma* vulnerabilidade em uma solução implantada. Por exemplo, as proteções, como o firewall baseado em host, podem significar que um sistema específico não esteja vulnerável a certas exposições classificadas como "altas".

[13] As informações são obtidas são tiradas da documentação oficial do
ICAT, encontrada em http://icat.nist.gov/icat_documentation.htm

[14] A Oracle publicou um mapeamento dos nomes CVE que foram externamente atribuídos a algumas vulnerabilidades listadas nos alertas de segurança em
http://www.oracle.com/technology/deploy/security/pdf/public_vuln_to_advisory_mapping.html. Esses nomes
CVE representam apenas um subconjunto dos problemas e não há mapeamento direto de nomes CVE para as vulnerabilidades específicas enumeradas nos boletins.

Enquanto as classificações contextuais podem apresentar a priorização da implantação de patches em uma organização, é importante lembrar que as configurações estão constantemente em fluxo e a proteção contextual é apenas temporária. As vulnerabilidades devem ser fixadas em sua raiz para realmente limitar a exposição durante o tempo. Outro aspecto a ser considerado é que uma simples vulnerabilidade pode ser aplicada a múltiplos sistemas (como múltiplas versões do Windows ou distribuições do Linux) que isso pode ter uma severidade contextual diferente para uma versão particular. Quanto a isso, a severidade baseada em CAN pode ser um fraco instrumento para que os administradores analisem o impacto de uma vulnerabilidade em uma aplicação específica /versão de OS ou sua implantação particular do sistema.

OBSERVAÇÃO da Análise: Após termos publicado o estudo sobre a Função do Web Server, uma das questões que recebemos foi sobre a razão de não termos feito uma análise de severidade pelos sistemas de classificação do fornecedor, em vez das classificações do ICAT. Há diversas razões pelas quais o ICAT é a melhor escolha para a análise de severidade.

Primeiro, o Red Hat não forneceu classificações de severidade durante todo o período de tempo estudado na primeira vez (o ano de 2004 inteiro), portanto, isso não seria possível. O Red Hat apresentou classificações de severidade às suas pesquisas em Fevereiro de 2005.

Em segundo lugar, embora os rótulos de classificação da Microsoft e da Red Hat pareçam iguais, não está claro ainda que eles sejam exatamente comparáveis ou se são aplicados da mesma maneira.

Por fim, queremos que nossa metodologia controle outros fornecedores. Mesmo que os sistemas de classificação da Microsoft e do Red Hat tenham sido idênticos, a Oracle não os utiliza, assim como a Apple também não, ou a Novell etc.

Em resumo, usamos o ICAT, pois precisávamos de uma classificação objetiva e que se aplicasse aos fornecedores e produtos.

Outra questão interessante que recebemos durante a primeira verificação da metodologia foi "por que não usamos a métrica de severidade do CERT?" Basicamente, a métrica de severidade do CERT é subjetiva e poderia mudar quase que diariamente, com base em sua definição. Para outros detalhes, confira nossa observação no relatório de Função da Web[1].

[1] Relatório de análise da Security Innovation, "Relatório de Comparações entre Funções – Função do Web Server," Março de 2005,

Análise da Função do Servidor de Banco de Dados

A seção que segue descreve, em detalhes, os passos desempenhados para configurar diferentes funções. Eles são apresentados para que nossos dados e resultados sejam reproduzidos por terceiros.

MySQL no Red Hat Enterprise Linux

Visão Geral da Instalação

A instalação da plataforma Red Hat Enterprise Linux é conduzida por assistente e fácil de ser executada. Para alcançar uma instalação mínima da função do servidor de banco de dados com o MySQL, nós desmarcamos a seleção de todas as opções do pacote de instalação.

E depois selecionamos manualmente apenas o grupo do pacote 'mysql'. Isso representa a instalação mínima do Red Hat configurável por meio das principais opções do assistente de instalação, que permite que o servidor funcione na função do Servidor de Banco de Dados. Durante a instalação, a senha de boot está desativada por padrão.

O Mysql-server não está atualmente nos CDs do RHEL, portanto, baixamos este módulo e o instalamos após completar o restante da instalação.

Recursos consideráveis de segurança

Durante a instalação da plataforma, um firewall é instalado e não permite tráfego no sistema, por padrão. O firewall é um simples firewall de entrada, bloqueando o tráfego de chegada ao sistema, mas sem fazer nada para a saída. O software `up2date`, que baixa automaticamente as atualizações para o sistema, está instalado por padrão. Durante a instalação, o assistente pergunta que tipo de tráfego deve ser desbloqueado pelo firewall.

Oracle 10g no Red Hat Enterprise Linux

Visão Geral da Instalação

A instalação da plataforma Red Hat Enterprise Linux é conduzida por assistente, sendo muito amigável. O Oracle apresentou diretrizes precisas de instalação recomendadas para o 10g no Red Hat Enterprise Linux 3, que estão detalhadas no Apêndice B deste relatório. É interessante notar que o sistema resultante possui muito mais componentes de OS dos suplementos do que a construção do MySQL. O sistema do Oracle, por exemplo, inclui o X Windows (Gnome), o Mozilla e muitos outros pacotes adicionais.

Recursos consideráveis da segurança

Durante a instalação da plataforma, um firewall é instalado e não permite tráfego no sistema por padrão. As diretrizes de instalação do Oracle recomendam desativar o firewall durante o procedimento de instalação, para fins de teste. O software do `up2date`, que baixa automaticamente as atualizações para o sistema, é instalado por padrão. Enquanto o up2date gerencia os patches suportados pelo Red Hat, os administradores ainda gerenciam as atualizações do Oracle.

SQL Server 2000 no Windows Server 2003:

Visão Geral da Instalação

A instalação do Windows Server 2003 é conduzida por assistente, sendo muito objetiva. A configuração padrão não instala aplicativos supérfluos. Ao final da instalação, outro assistente automaticamente notifica o usuário sobre a configuração de uma função para o servidor.

Recursos consideráveis da segurança

O firewall é instalado, mas deve ser habilitado, e então não permite tráfego no sistema. O SQL Server 2000 comunica o TCP/IP usando a biblioteca de rede de sockets. O serviço automático de atualização do Windows é instalado e executado por padrão. Desde a data de publicação deste relatório, essa ferramenta agora gerencia as atualizações às aplicações instaladas no Windows Server 2003, incluindo o SQL Server 2000.

Resultados da Contagem de Vulnerabilidades

Apenas o Software de Banco de Dados

Antes de analisar o funcionamento geral dos servidores de banco de dados, vamos primeiro examinar o software de banco de dados separadamente.

SQL Server 2000 SP3

Nos 12 meses estudados, o SQL Server 2000 contribuiu com zero vulnerabilidade para a função do servidor de banco de dados. Enquanto o SQL Server 2000 SP3 teve algumas vulnerabilidades antes do período de 12 meses, examinando o histórico do produto, descobrimos que ele possui o menor número de vulnerabilidades entre os bancos de dados estudados.

Uma vez que o SQL Server 2000 SP3 era vendido em Janeiro de 2003, houve então um total de cinco vulnerabilidades em torno de Fevereiro de 2005, o que fornece, em média, 0.2 vulnerabilidade por mês.

MySQL 3.23.58

Nos 12 meses estudados, o MySQL forneceu sete vulnerabilidades à função do servidor de banco de dados, o que representa o total encontrado no final de Fevereiro de 2005, uma vez que o RHEL 3 foi lançado em 23 de Outubro de 2003. Isso resulta em uma média de 0.4375 vulnerabilidade por mês, ou seja, duas vezes a média descoberta do SQL Server 2000 SP3.

PostgreSQL

Embora o postgreSQL não faça parte de nenhum dos três servidores analisados, como concorrente próximo do MySQL, no mundo do código aberto, é interessante analisar seu impacto potencial, uma vez que o selecionamos. Nos 12 meses estudados, o postgreSQL apresentou cinco vulnerabilidades diagnosticadas pelo Red Hat, e seis vulnerabilidades no total, até o final de Fevereiro de
2005, uma vez que o RHEL 3 foi lançado em 23 de Outubro de 2003. O resultado de um ano foi uma média de 0.417 vulnerabilidade por mês, muito semelhante à taxa do MySQL.

Oracle 10g

Nos 12 meses estudados, o Oracle 10g contribuiu com 30 vulnerabilidades para a função do servidor de banco de dados, ou de aproximadamente 2,5 por mês, o que também representa o total encontrado no final de Fevereiro de 2005, uma vez que o Oracle 10g foi lançado em Fevereiro de 2004. Este resultado fornece uma média de 2.3 vulnerabilidades por mês, uma taxa um pouco maior do que a MySQL ou SQL Server 2000.

Houve ainda alguns outros desafios referentes à análise das vulnerabilidades do Oracle 10g, se comparado ao Red Hat ou Microsoft. As 30 vulnerabilidades ajustadas no Oracle 10g foram identificadas nos alertas de segurança (encontre-os em http://www.oracle.com/technology/deploy/security/alerts.htm), da Oracle, conforme segue:

Alerta 68, Atualização de Segurança Oracle identificou as vulnerabilidades que afetaram o 10g, como (usando dados da Oracle) DB01, DB02, DB05, DB08, DB09, DB10, DB11, DB12, DB13, DB14, DB15, DB16, DB20, DB22, DB24, DB25, DB26, DB27, e DB28, para um total de 19 vulnerabilidades solucionadas.

Atualização Crítica de Patch, em Janeiro de 2005, identificou vulnerabilidades que afetaram o 10g, como DB03, DB06, DB07, DB08, DB09, DB11, DB12, DB13, DB14, DB15 e DB16, para um total de 11 vulnerabilidades solucionadas.

Esses alertas não fornecem informações sobre quais identificadores CVE se associam aos identificadores da Oracle, no entanto, a Oracle fornece um documento de mapeamento em http://www.oracle.com/technology/deploy/security/pdf/public_vuln_to_advisory_mapping.html. Este documento mapeia os identificadores CVE apenas nos documentos de Alerta da Oracle, e não às vulnerabilidades individuais, mapeando apenas um total de dez CVEs, com exceção das 30 vulnerabilidades identificadas e solucionadas no Oracle 10g.

Após uma verificação extensiva, não podemos afirmar de que forma os CVEs mapeiam os identificadores individuais do DB*NN* nos Alertas. É provável que alguns pesquisadores publiquem, separadamente, detalhes no Bugtraq ou em outras fontes, depois que a Oracle emite um patch, o que resulta na atribuição das IDs do CVE, mas, para outros problemas, isso não aconteceu. Para fins de análise no estudo, supomos o seguinte:

As vulnerabilidades DB*NN* que não possuem um identificador CVE também foram encontradas pela Oracle ou reportadas sob a divulgação responsável da Oracle, para se obterem soluções, uma vez que não conseguimos encontrar outras abordagens sobre elas.

As vulnerabilidades, sem um identificador CVE, serão consideradas descobertas na data em que a Oracle emitiu seu Alerta, contribuindo com zero dia de risco.

Assemelha-se então a uma área em que a Oracle possa fazer algumas melhorias para benefício dos consumidores, participando do processo Mitre CVE. Sem a atribuição de um identificador CVE, os produtos de Detecção e Limpeza de Intrusos, geralmente usados, compatíveis com CVE, não necessariamente encontrarão e reportarão as vulnerabilidades.

Por outro lado, essa taxa de complexidade parece dificultar também para os atacantes, uma vez que poucos detalhes foram descobertos a respeito das vulnerabilidades.

Função do Servidor de Banco de Dados – OS do Servidor e Software de Banco de Dados

A tabela que segue resume nossas descobertas com relação à contagem de vulnerabilidades da instalação analisada:

Severidade	Windows Server 2003/SQL Server 2000	RHEL ES 3/Oracle 10g	RHEL ES 3/MySQL 3.23.58
Alta	27	73	41
Média	18	63	49
Baixa	0	10	8
Desconhecida	18	61	18
Total	63	207	116

Tabela 1: Contagem de vulnerabilidades para a comparação de três soluções
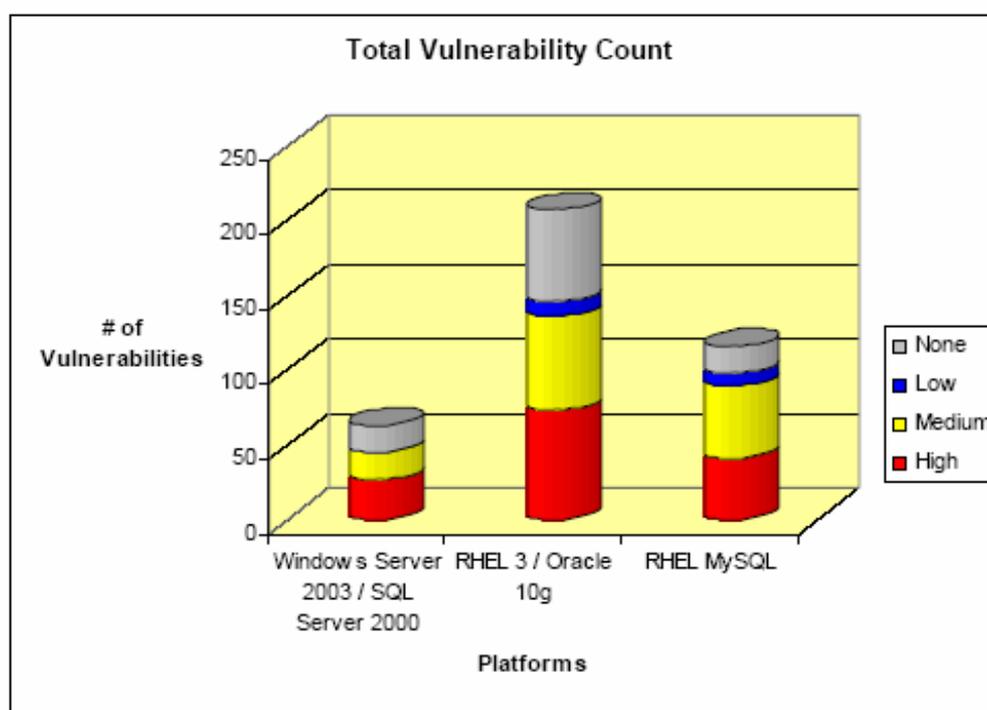
Figura 1: Contagem de vulnerabilidades para 2004, considerando-se três configurações.

Esta tabela inclui as vulnerabilidades de todos os softwares instalados na plataforma pelos procedimentos de instalação descritos na seção anterior. A Figura 1 representa graficamente esses resultados. No caso da solução do MySQL RHEL, fizemos um grande levantamento da modularidade do Linux para criar um sistema com componentes minimamente instalados. Os procedimentos de instalação sugeridos pela Oracle, por outro lado, favorece a riqueza do recurso ao solicitar uma interface gráfica composta por janelas, e sugerindo a instalação dos pacotes do Red Hat, como as “Ferramentas do Desenvolvedor” e outras que expandem a funcionalidade do servidor, ao mesmo tempo em que aumenta a superfície de ataque do sistema. No SQL Server 2000, do caso do Windows Server 2003, consideramos uma instalação completa de todos os componentes que acompanham a plataforma.

A solução do Windows trouxe mais de 63 vulnerabilidades se comparado às 116, no sistema do MySQL RHEL, do Oracle RHEL. No caso do Windows, levamos em conta uma instalação completa do sistema operacional base, o que significa que contamos todas as vulnerabilidades dos componentes do Windows Server, estivessem ou não instalados. Para os outros casos, contamos apenas aquelas vulnerabilidades que afetam os pacotes instalados nesses sistemas.

Uma olhada mais detalhada nos dados revela que o Internet Explorer tinha mais vulnerabilidades do que quaisquer outros componentes do Windows, em seu período de estudo, chegando a 16 dos 63 problemas de segurança solucionados no SQL Server 2000, na solução do Windows Server 2003. A Microsoft
Pode se beneficiar com uma perspectiva da superfície de ataque, tornando seus futuros sistemas operacionais de servidor mais modulares, para que os usuários possam remover esses componentes, se desejado. Levando em conta que, aproximadamente, 30% das vulnerabilidades solucionadas em um período de um ano de análise no Windows Server 2003 estavam relacionadas ao Internet Explorer, fornecer aos administradores a opção de remover tais componentes iria reduzir a contagem da vulnerabilidade para o Windows Server 2003 ainda mais.

Para as duas projeções baseadas no Linux, o Kernel foi o maior colaborador à contagem total de vulnerabilidades, com 38 reportadas em nosso período de estudo, afetando tanto o MySQL, na solução do RHEL (38 das 116 vulnerabilidades estavam no kernel) como a solução RHEL (38 das 207 vulnerabilidades estavam no kernel). Na construção do Oracle, depois do kernel do Linux, o software Oracle 10g contribuiu com o próximo maior número de vulnerabilidades, com 30 no total, durante o tempo estudado.

Deveria ter sido observado que o Red Hat Enterprise Linux ES 3 é modular em seu procedimento de instalação, podendo funcionar com alguns dos componentes que são “principais” no sistema do Windows. Isso é potencialmente significativo, pois, um menor número de componentes instalados significa uma superfície de ataque reduzida e menos patches que podem ser aplicados. Também deveria ser observado que a superfície de ataque do Oracle pode também ser reduzida, removendo-se alguns componentes sugeridos em seus procedimentos de instalação.

A análise de vulnerabilidade mostra uma vantagem à plataforma do Windows Server 2003, levando em consideração diversas métricas importantes. As contagens totais de vulnerabilidades da plataforma Windows são significativamente menores que aquelas para as soluções do Oracle e do MySQL, projetadas no RHEL ES 3; da mesma forma, ao analisar apenas os bugs de severidade alta, as contagens são claramente a favor da Microsoft.

Sem uma metodologia rigorosa, pode-se esperar que uma mínima parte do MySQL, no Red Hat Enterprise Linux, tenha a menor quantidade de vulnerabilidades, seguida do Oracle, no Red Hat Enterprise Linux e o SQL Server 2000 no Windows Server 2003. No entanto, os verdadeiros resultados demonstram que o Microsoft Security Development Lifecycle (SDL) resulta em um software com menor taxa de incidência a vulnerabilidades, contribuindo com menos vulnerabilidades e menos patching para os clientes do que o Oracle ou as alternativas de Código Aberto estudadas.

Embora o foco do estudo esteja nas 3 configurações descritas, já antecipamos que alguns leitores podem ainda questionar o seguinte:

E o Oracle, em certa parte do Linux, semelhante à construção do MySQL?

E o Oracle, quando instalado no Windows Server 2003, em vez do Linux?

A Figura 2 do gráfico mostra variações nas construções que estudamos e mostra as respostas a essas questões. O servidor de banco de dados é projetado com as mínimas vulnerabilidades juntamente com as maiores vulnerabilidades, no período de 12 meses, da seguinte forma:

1. (mínima) SQL Server 2000 SP3 no Windows Server 2003
2. Oracle no Windows Server 2003
3. MySQL no Red Hat Enterprise Linux 3 ES (construção mínima)
4. Oracle no Red Hat Enterprise Linux 3 ES (construção mínima)[15]
5. (maior) Oracle no Red Hat Enterprise Linux 3 ES (Instruções de instalação do Oracle). Esses resultados mostram que o Oracle, na solução Windows, apresentou menos vulnerabilidades que o Oracle na solução Linux, mas que o SQL Server no Windows apresentou menos vulnerabilidades por uma margem significativa.
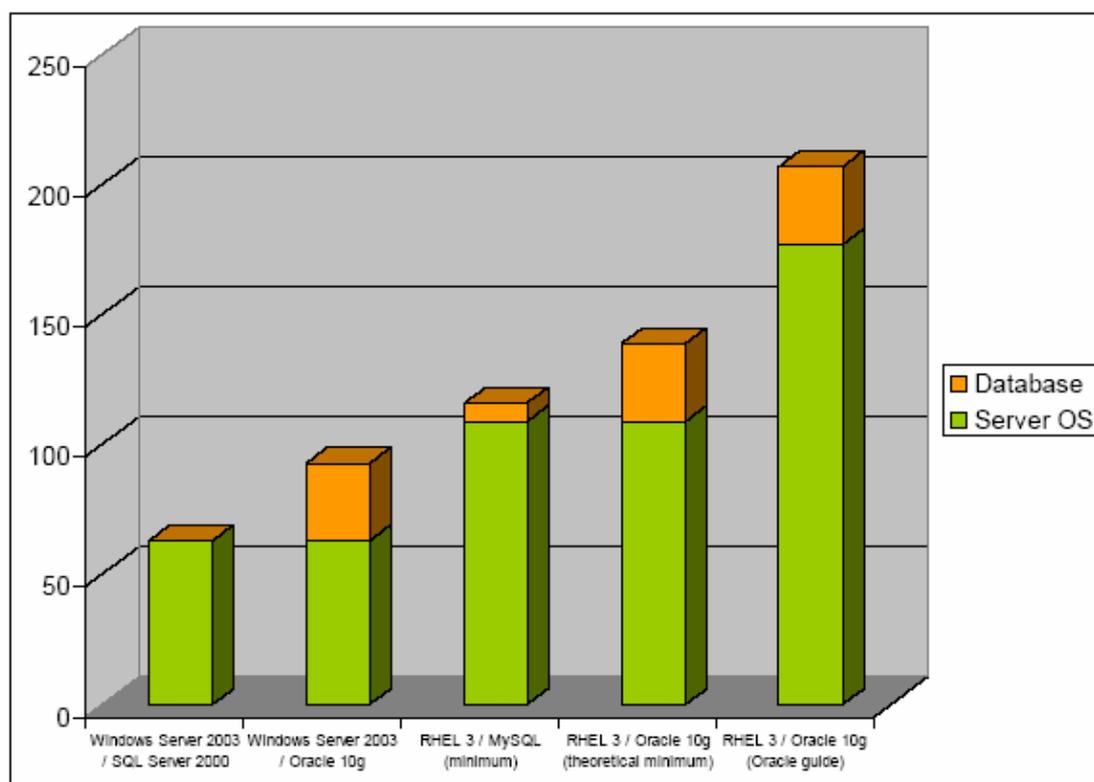
Figura 2: Variações do Banco de Dados e Plataforma

[15] Os números do Oracle para a configuração mínima do Red Hat presumem que possamos executar o Oracle efetivamente após remover os componentes recomendados de instalação. O verdadeiro número de vulnerabilidades é, portanto, um pouco maior do que os dados mostrados no gráfico apresentado.

Discussão sobre Dias de Risco

Cumulativos e Médios

Em nossas métricas, referimo-nos a dias de risco cumulativos e médios. Cada um deles é importante de ser considerado, pois oferece alguma idéia na segurança central do usuário quanto à solução, e também um serviço de segurança ao fornecedor. *Dias de risco cumulativos* referem-se à exposição geral de uma solução, assim como o serviço de segurança do fornecedor em certa estatística. Especificamente, os dias de risco cumulativos interrompidos pela severidade Isso se refere ao número total de dias de exposição – o que pode envolver diferentes vulnerabilidades – quando o sistema esteve no auge de seu risco por meio da exploração de uma vulnerabilidade. Comparando os números, vemos que o SQL Server 2000 na solução do Windows Server 2003 apresentou 952 dias de risco cumulativos para vulnerabilidades com classificação ICAT "alta", comparando-se com as 1525 do My SQL na solução do RHEL, e 2539 do Oracle na solução do RHEL, demonstrando clara liderança da Microsoft.

Os dias de risco médios representam uma medida valiosa da resposta de segurança do fornecedor, uma vez que se refere ao tempo médio entre quando a vulnerabilidade é divulgada e quando um fornecedor a conserta, tornando-a disponível. Os números do período considerado mostram uma liderança da Microsoft nesta categoria, assim como uma média de 32.0 dias, representando o período de tempo que os clientes são expostos a níveis maiores de risco por vulnerabilidade, se comparados com a média de 61.6 dias de risco para o MySQL na solução do RHEL e 38.7 dias de risco do Oracle na solução RHEL. É interessante observar como esses dados refletem a descoberta responsável de vulnerabilidade que a Microsoft promove de forma ativa, levando diversas vulnerabilidades a zero dia de risco, o que significa que a vulnerabilidade é descoberta no mesmo momento em que a solução é disponibilizada. Observe que a utilização extensiva do Oracle, a respeito da divulgação, também faz com que a construção do Oracle sobre o RHEL tenha uma média menor de dias de risco do que o MySQL no RHEL, embora o maior número total de vulnerabilidades ainda leve a dias de risco cumulativos maiores. Uma estatística comprovada é que os dias médios de risco para vulnerabilidades, na solução da Microsoft, são 0, ou seja, o oposto dos 23 dias de risco do MySQL na solução do RHEL, e 18 para o Oracle no RHEL.

Um aspecto interessante do desafio que o Red Hat enfrenta, que não é tão óbvio a partir de uma simples análise dos números brutos, é a demora entre a disponibilidade de uma solução dentro do produto e a inclusão do produto como um pacote Red Hat "aprovado". Por exemplo, o CAN-2004-0836 trata de um bug na função mysql_real_connect() do MySQL. Isso foi registrado no banco de dados de bugs do MySQL em 4 de Junho de 2004, e solucionado na raiz em 17 de Junho de 2004. No entanto, o Red Hat disponibilizou essa solução apenas na RHSA - 2004:611, em 27 de Outubro. Esse problema de gerenciamento de soluções publicadas a partir de terceiros é muito complicado, podendo representar um desafio significativo ao Linux, em se tratando de uma base contínua.

OBSERVAÇÃO da Análise: Um dos principais itens que aprendemos durante o estudo dos nossos resultados preliminares foi que existem fortes opções de valor, ou falta de valor, para as métricas dos dias de risco. Para nós, vemos muito claramente a distinção no risco do cliente, tanto antes como depois de um evento de tornar publicamente conhecido. Dessa forma, os dias de risco são uma medida interessante para se analisar como a combinação das políticas de divulgação do fornecedor, o processo de resposta e os testes e liberações de patches se combinam para reduzir (ou não) o período quando os clientes estão expostos sem uma alternativa de patch do fornecedor. É uma medida do mundo real para um problema do mundo real, e, queira ou não, ela é afetada pela forma com que o software é desenvolvido.

Um dos pontos levantados foi o tipo de métrica que automaticamente favorece uma solução de código fechado. Outro ponto questiona a defesa da Microsoft quanto à "divulgação responsável", uma vez que ela serve para tornar menores os dias de risco e fazer a Microsoft analisar melhor. Na verdade, um fornecedor com processos maiores de testes de qualidade pode se beneficiar mais da divulgação responsável, mas, uma vez que é a política que conduz o menor risco ao cliente, isso parece intensificar, e não reduzir, a importância da métrica.

Na verdade, os dados são tão claros que as comunidades da Microsoft e do Linux seguem a divulgação responsável a certo nível. Por exemplo, para o problema do Samba, solucionado pelo Red Hat na RHSA-2004:670-10, pode-se seguir as referências para saber que os fornecedores mantiveram o problema longe de ser divulgado, na data em que foram notificados pela primeira vez. Segue um registro parcial do bugzilla:

> *Jerry no Samba reportado ao fornecedor-sec em 2004120, uma falha de raiz remota no Samba, afetando todas as versões. Requer usuário autenticado. Problema descoberto pela iDEFENSE.*
>
> *Este problema está atualmente inativo até 20041216:1200UTC*

Pode-se pensar que este problema era "público" logo que foi reportado ao fornecedor-sec, devido ao número de pessoas na lista. No entanto, usamos 16/12/2004 como a primeira data publicamente conhecida. O Red Hat apresentou 15 problemas solucionados com zero dia, e a maioria beneficiou, com a divulgação responsável, de forma privada, aos fornecedores do Linux – as ações foram elogiadas.

Além disso, parece estar claro que o risco do cliente poderia ser reduzido ainda mais por uma coordenação maior da segurança pelos fornecedores de Linux. Observe o exemplo da RHSA-2004:413-07, que diagnosticou um problema de kernel e o tornou público em 3/8/2004. Os usuários do SuSE não obtiveram diagnóstico até 9/8/2004 e os usuários do Gentoo Linux esperaram até 25/8/2004, de acordo com referências da lista do CVE.

As tabelas que seguem mostram os resultados obtidos para as análises dos dias de risco. Tanto os dias de risco cumulativos como os médios são calculados, pois representam métricas diferentes sobre as vulnerabilidades e a resposta de patching.

Severidade	Windows Server 2003/SQL Server 2000	RHEL ES 3/Oracle 10g	RHEL ES 3/MySQL 3.23
Dias de Risco: Severidade Alta	952	2539	1525
Dias de Risco: Severidade Média	573	3314	3594
Dias de Risco: Severidade Baixa	0	574	741
Dias de Risco: Desconhecida	490	1590	1290
Dias de Risco Cumulativos	2015	8017	7150
Dias de Risco Médios por Vulnerabilidade	31.98	38.73	61.64

Tabela 2: Dias de risco nos três sistemas estudados
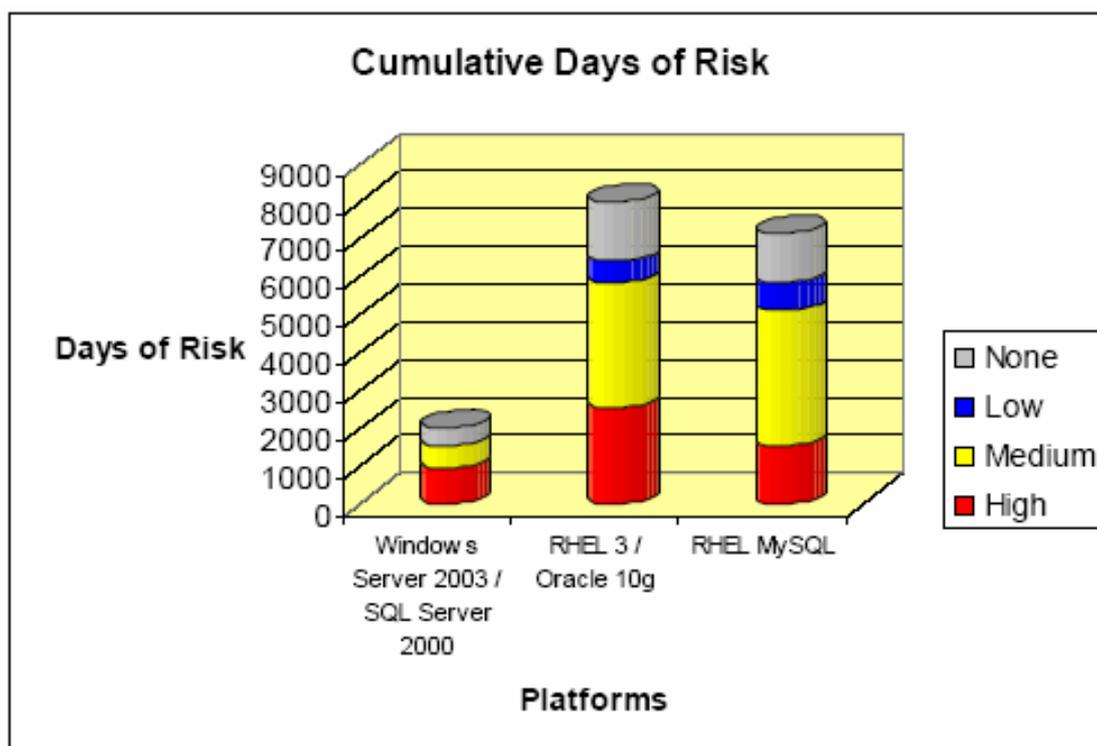
Figura 3 – Dias de risco cumulativos para os três sistemas estudados
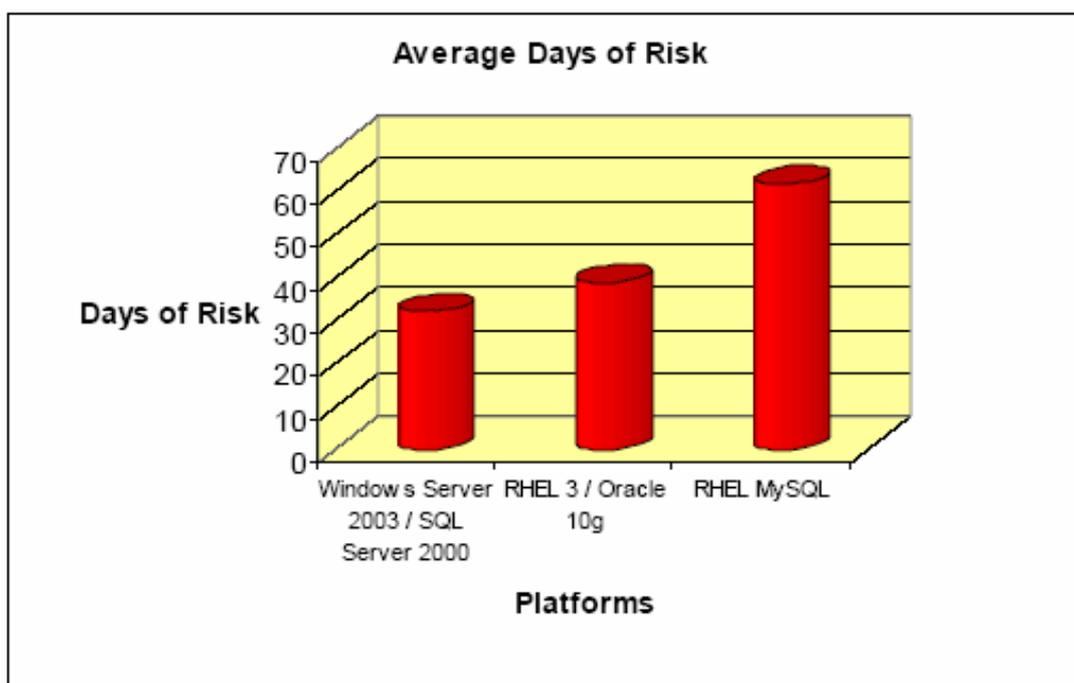
Figura 4 – Dias de risco médios para os três sistemas estudados

Distribuição de Dias de Risco

É interessante analisar além dos dias de risco médios e examinar a proporção de vulnerabilidades solucionadas dentro de certo tempo. As Figuras 4 – 6 explicam as vulnerabilidades solucionadas em 2004 nas plataformas/configurações consideradas categoricamente pelos dias de risco. Em todas as plataformas, a proporção de vulnerabilidades, nas categorias de dias de risco de 0-30 dias, é maior. Analisando os dados, revela-se que 87.2% das vulnerabilidades da categoria 0-30 da Microsoft é de zero, opondo-se aos 24.3% zeros do MySQL, na solução do RHEL, e 41.1% do Oracle na solução RHEL, dentro desta categoria. Isso se refere às diferenças no modelo de divulgação das duas plataformas. Essa distribuição é importante, como a pesquisa sobre a taxa de exploração às vulnerabilidades mostrou um grande aumento na taxa de exploração, à medida que o tempo passava a partir da divulgação. Dessa forma, as vulnerabilidades com maior vida útil têm um impacto desproporcional na segurança geral da plataforma, além de terem efeito nos cálculos dos dias de risco.
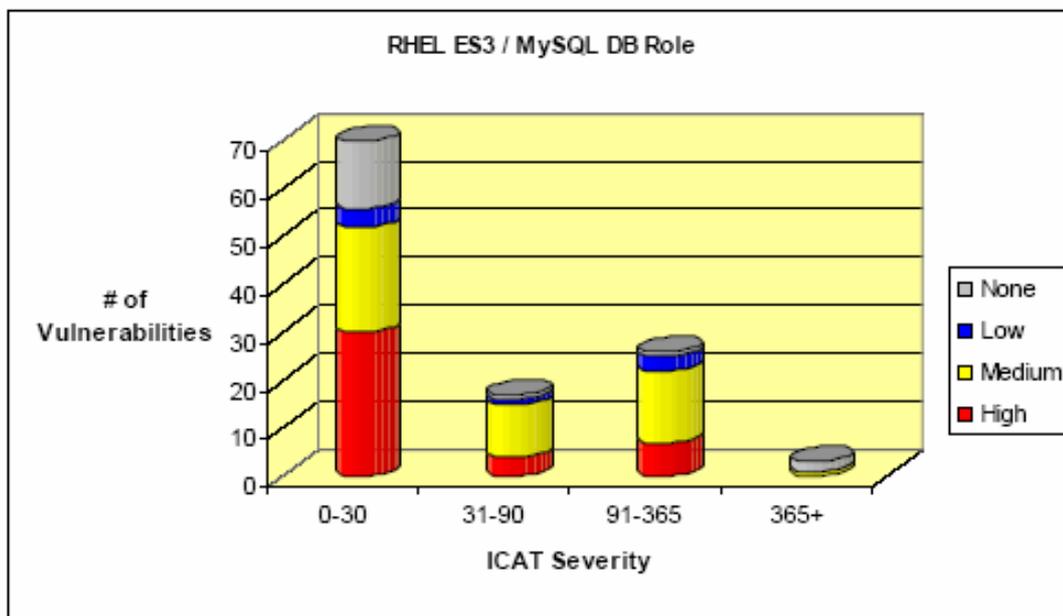
Figura 5: Dias de Risco do RHEL 3 com o MySQL apresentados pelo tempo de solução
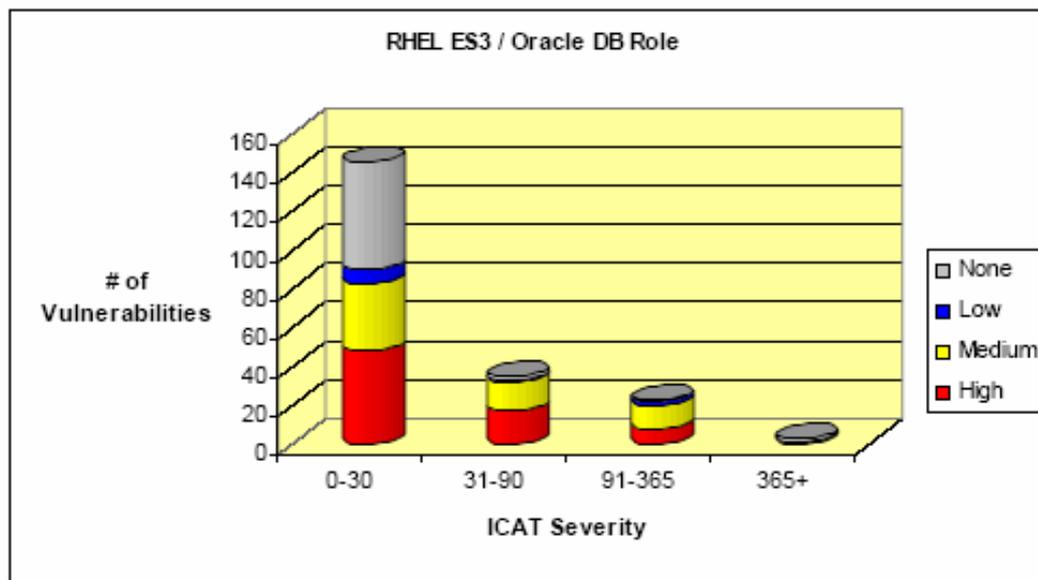
Figura 6: Dias de Risco do RHEL 3 com o Oracle apresentados pelo tempo de solução
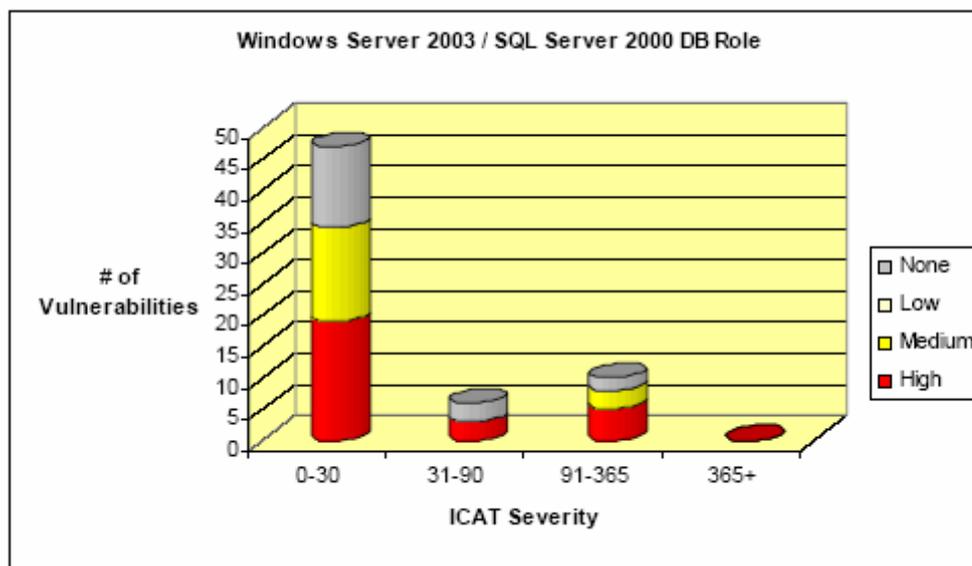
Figura 7: Dias de Risco do SQL Server 2000 no Windows apresentados pelo tempo de solução

Visão Detalhada dos Ajustes Com Prazo de +90 Dias

As três configurações consideradas possuem certas vulnerabilidades que tinham mais de 90 dias entre sua descoberta e a liberação de uma solução. Abaixo, iremos detalhar essas vulnerabilidades para obter soluções analisadas.

SQL Server 2000 no Windows Server 2003 – Existem dez vulnerabilidades solucionadas no período de um ano, sob a consideração de que há mais de 90 dias de risco e, destas, cinco foram designadas pelo ICAT como sendo de severidade alta. Das vulnerabilidades que restaram, três foram designadas como de média severidade e duas não passaram pelo processo de classificação no momento da análise. Cinco dessas vulnerabilidades, CAN-2005-0053, CAN-2004-0727, CAN-2004-0841, CAN-2003-1041 e CAN-2003-1048, estavam no navegador do Internet Explorer, quatro estavam no sistema principal e uma no ASP .NET.

MySQL no RHEL – Houve vinte e nove vulnerabilidades solucionadas no período de um ano, levando-se em conta mais de 90 dias de risco e, delas, sete foram designadas pelo ICAT como sendo de severidade alta, dezesseis como média, três como baixa e três ainda não passaram pelo processo classificatório no momento do estudo. Doze dessas vulnerabilidades estavam no kernel do sistema operacional. O restante foi distribuído entre uma variedade de pacotes, com o MySQL como o segundo maior ofensor (atrás do kernel), com sete vulnerabilidades.

Oracle no RHEL – Houve vinte e seis vulnerabilidades solucionadas em 2004, em mais de 90 dias de risco, e, delas, oito foram designadas pelo ICAT como de severidade alta, treze como média, duas como baixa e três ainda não receberam a classificação. Doze dessas vulnerabilidades estavam no kernel do sistema operacional, com o restante distribuído entre uma variedade de pacotes.

Critérios Qualitativos da Segurança

Enquanto a análise das vulnerabilidades e liberação dos patches nos fornece uma idéia quanto à "capacidade de exploração" de um sistema, existem fatores adicionais que podem ser importantes para aqueles que desejam tomar decisões sobre o desenvolvimento e a implantação de uma plataforma. Esses critérios irão ajudar a tomar decisões sensíveis de segurança, no contexto de considerações operacionais práticas, que podem ser importantes a um departamento de TI. Mais especificamente, fazemos a comparação entre duas plataformas - Red Hat Enterprise Linux 3 e Windows Server 2003 – que serviram como base para as três configurações estudadas neste relatório.

De interesse particular, estão os recursos de segurança que são parte padrão da plataforma. Por exemplo, os clientes prestam muita atenção nos recursos de autenticação, suporte a VPNs, capacidade de Atualização automática, proteção ao estouro de buffer, recursos gerenciados de código e trilha de auditoria. Alguns dos problemas qualitativos mais importantes são relacionados nas seções que seguem; essa lista não é exaustiva e não deve ser compreendida como uma relação abrangente dos recursos referentes à segurança. No lugar disso, ela destaca os recursos que estão prontamente medidos por um cliente bem informado.

Proteção da Porta/Firewall

Ambos os fornecedores oferecem suporte a um firewall por padrão. Os dois aplicativos de firewall são básicos e baseados em tabelas de IP. O firewall é instalado por padrão nas duas plataformas, bloqueando todas as solicitações de chegada, quando executado. Para o Red Hat Enterprise Linux 3, o firewall está habilitado por padrão. Já no Windows Server 2003, ele deve ser manualmente ativado.

O ICF (Firewall de Conexão à Internet), na plataforma do Windows Server 2003, desempenha funções básicas, podendo ser configurado para bloquear com sucesso as conexões ou pacotes com falhas. As configurações do ICMP podem ser usadas para permitir que o servidor comunique as informações de status da rede. As configurações padrões do ICMP desabilitam a maioria das comunicações do ICMP, exceto as solicitações de eco de ICMP de saída.

O software `iptables,` no servidor do Red Hat, possui uma interface baseada em texto ("Nível de Segurança"), que permite a configuração básica do firewall (i.e. habilitar ou desabilitar o firewall, permitindo conexões para um número restrito de serviços). Essa interface também não permite que as configurações do ICMP sejam alteradas (as respostas do eco do ICMP são habilitadas por padrão) ou monitorem os logs do firewall. O recurso mais importante é a filtragem de nível de pacote, que permite que o administrador estabeleça regras de firewall em qualquer aspecto do pacote. As opções de linha de comando também permitem o bloqueio do tráfego com base na associação de pacotes a uma determinada regra.

Diretiva de Suporte ao Ciclo de Vida

Um dos aspectos da segurança é a duração do serviço de suporte. Se um produto não é mais suportado, os usuários são forçados a atualizar ou encarar riscos avançados de uma brecha na segurança. Enquanto as informações aqui são atualizadas a partir da data de publicação deste relatório, as diretivas de suporte se desenvolveram de forma significativa durante os últimos anos, para os dois fornecedores com a probabilidade de continuarem no fluxo.

Então, sugerimos que você visite o site do fornecedor para obter as informações mais atualizadas.

A tendência dos servidores nos últimos anos, levando-se em conta os requisitos do cliente, é de ampliar o ciclo de vida padrão. O Red Hat 9 durou apenas um ano, mas, com as versões Empresariais, o Red Hat introduziu um compromisso para o ciclo de vida do suporte de 7 anos de segurança[16]. Da mesma forma, em 2002, a Microsoft padronizou suas diretivas de suporte, ampliando recentemente esse ciclo de vida para 10 anos[17]. Começando com sua versão de Servidor de Banco de Dados 9i, a diretiva de "Suporte à Correção de Erros" da Oracle foi estendida para cinco anos a partir da liberação do produto, com uma notificação de 12 meses para os clientes, antes de descontinuar o suporte[18].

O suporte da Microsoft para o Windows Server 2003 começou na data de lançamento, em 28 de Maio de 2003, e está planejado para continuar até Maio de 2013. Seu suporte ao SQL Server 2000 começou com o lançamento do produto, em 30 de Novembro de 2000, planejado para continuar por sete anos a partir da próxima versão se seu servidor de banco de dados (previsto para 2005), com suporte até 2012. O suporte do Red Hat para o Enterprise Linux 3.0 começou na data de lançamento – 23 de Outubro de 2003 – planejado para ser mantido por 7 anos, terminando em 2010. O Suporte à Correção de Erros do Servidor de Banco de Dados da Oracle 10g começou em seu lançamento, no início de 2004, e está programado para continuar até 2009.

Em resumo, os fornecedores se comprometeram a, pelo menos, cinco anos de suporte à segurança em longo prazo para seus softwares de servidor. Os tomadores de decisão devem analisar as implicações dessas diretivas, à medida que elas se relacionam com suas necessidades de produção.

Boletim/Descrição Informativa

Os boletins de segurança representam geralmente a única parte da informação que os administradores de sistema utilizam para tomar decisões a respeito da implantação de patch a partir de uma perspectiva de gerenciamento de riscos. Portanto, é importante que as pesquisas contenham informações para que essas pessoas sejam capazes de tomar decisões bem informadas e contextualmente relevantes.

As pesquisas do Red Hat (veja https://rhn.redhat.com/errata/rhel3as-errata-security.html) são muito sucintas e contêm uma breve descrição sobre as vulnerabilidades solucionadas.. A pesquisa contém pouca informação sobre o contexto em que uma vulnerabilidade está presente. Não há informações sobre patch, a não ser que o número da versão do arquivo/número do patch seja fornecido.

Os boletins de segurança da Microsoft contêm dados sobre os fatores solucionados, possíveis demonstrações, conseqüências do patching e uma descrição do processo de patching (veja http://www.microsoft.com/technet/security/current.asp para conferir os Boletins de Segurança da Microsoft). As pesquisas contêm também informações sobre o escopo e as conseqüências da vulnerabilidade, incluindo uma extensão aos danos que podem ocorrer. Além disso, elas incluem informações sobre todos os arquivos que serão modificados.

---

[16] https://www.redhat.com/apps/support/

[17] http://support.microsoft.com/default.aspx?scid=fh;%5Bln%5D;LifeWin

[18] http://www.oracle.com/support/policies.html

Os Alertas de Segurança da Oracle foram aprimorados no ano passado, mas fornecem muito menos detalhes quanto às vulnerabilidades solucionadas. A Oracle fornece pequenos resumos das questões individuais de segurança solucionadas em formato tabular, fornecendo também uma instrução de "pré-condição para a capacidade de exploração 1-2" para cada vulnerabilidades. Um dos maiores desafios do cliente empresarial, no entanto, é ligar as informações dos boletins dos fornecedores com as fontes externas de informações que podem fornecer uma idéia sobre o perfil da ameaça e os verdadeiros riscos. Isso pode certamente ser aprimorado pela participação ativa do Oracle no programa Mitre CVE e sua inclusão nos nomes CVE para vulnerabilidades em seus boletins, como o Red Hat e a Microsoft fazem.

Divulgação do Impacto do Patch

Uma preocupação importante dos administradores de sistema é que um patch pode corromper a funcionalidade de outra aplicação, às vezes nem relacionada, que está sendo executada no sistema. Isso pode ser causado por incompatibilidades ou tempos de parada referentes a reinicializações durante o processo de instalação do patch. Algumas preocupações geralmente fazem com que os administradores atrasem a implantação de patches, enquanto sua validação ocorre, aumentando, portanto, o tempo durante o qual os sistemas estão vulneráveis, mas diminuindo as chances de o patch ter efeitos colaterais imprevistos.

Os boletins de segurança da Microsoft contêm informações referentes aos arquivos que serão modificados, às reinicializações e ao impacto de não se fazer o patching. Além disso, a Microsoft fornece ferramentas, como o Analisador de Segurança da Linha de Base, que determina se uma atualização é ou não solicitada para o sistema. No entanto, os boletins não contêm informações sobre a quantidade de tempo exigida para a instalação.

As pesquisas de segurança do Red Hat são orientadas pela vulnerabilidade; ou seja, as informações disponíveis referem-se à causa da vulnerabilidade e do perigo potencial que ela apresenta. As pesquisas não contêm nenhuma informação sobre o impacto do processo de patching, ou reinicializações exigidas. As únicas informações disponíveis, referentes ao patch, são as versões dos pacotes que estão implantados neste patch. O impacto *preciso* de um patch pode ser determinado pela análise das alterações feitas no código-fonte entre as versões. Enquanto isso é impraticável em muitos ambientes, é uma possibilidade adicional para os sistemas que são críticos.

As Atualizações Críticas de Patch da Oracle contêm informações sobre as quais o componente de alto nível é afetado pelas vulnerabilidades listadas, mas não fornecem indicações sobre o impacto do patch. Os clientes empresariais têm, portanto, a possibilidade de tentar determinar o impacto que o patch pode exercer para executar o sistema.

Tecnologia de Implantação do Patch

O processo adequado de patching das máquinas é um método poderoso de aprimorar a probabilidade de se permanecer seguro. Mesmo as ameaças mais atuais, que infectaram os CSOs do mundo todo, foram logo solucionadas por máquinas de patching rápidas e efetivas, uma vez que, em cada caso, os worms encontrados exploraram vulnerabilidades que ainda estavam sendo solucionadas pelas séries de patch atualmente disponíveis. Apesar do conflito que existe na maioria das organizações, entre a implantação automática de patch e o teste de compatibilidade de patches autorizados, a habilidade de as máquinas se atualizarem automaticamente é

Extremamente valiosa para os usuários individuais e os corporativos que não possuem recursos para suportar uma equipe de gerenciamento de patch.

A Microsoft e a Red Hat possuem recursos de atualização automática que permitem que um sistema obtenha, automaticamente, as atualizações funcionais e de segurança. Essas aplicações de auto-atualizações (Atualização Automática do Windows para o sistema operacional da Microsoft e o `up2date` para o da Red Hat) podem ser definidas para notificar e baixar, ou para notificar, baixar e instalar automaticamente os patches.

Uma grande vantagem da aplicação do `up2date` sobre a atualização automática do Windows é de que ele pode rastrear um software não OS através de canais de assinatura na Rede do Red Hat. No entanto, a atualização automática do Windows agora também atualiza aplicações instaladas no sistema operacional, incluindo o SQL Server 2000. Além disso, a Microsoft possui outras ferramentas disponíveis para o gerenciamento de patch que pode controlar outros produtos Microsoft em uma ferramenta centralizada (SUS/WUS – Software Update Services/Windows Update Services – e SMS 2003 – System Management Server 2003), permitindo maior flexibilidade.

Com o lançamento do 10g, a Oracle oferece sua ferramenta do Enterprise Manager, que automatiza a notificação e entrega dos patches da Oracle através de uma conexão com um site de suporte do Oracle's MetaLink. O Enterprise Manager também pode ser configurado para notificar os administradores sobre a disponibilidade do patch com base em seus produtos implantados e versões, permitindo que os administradores programem implantações de forma centralizada.

Liberação do Patch (Agrupamento)

Os três fornecedores adotaram uma diretiva de patches agrupados para liberação. A diretiva da Microsoft agrupa patches, em um ciclo mensal, a Oracle libera suas Atualizações Críticas de Patch a cada três meses e a Red Hat produz "erratas" sempre que possível, mas geralmente libera patches de segurança conforme o necessário.

Mais especificamente, a diretiva da Microsoft é de liberar patches na primeira Terça-feira de cada mês, a da Oracle é liberar suas atualizações críticas na Terça-feira próxima ao 15° dia de Janeiro, Abril, Julho e Outubro. Embora a Red Hat afirme que há uma liberação agrupada de patches, não existe nenhuma indicação de dados sólidos ou ciclo de liberação que permita que o administrador do sistema programe os ciclos de gerenciamento do patch.

Capacidade Reversa do Patch

No caso de um patch "ruim" que devasta um sistema ou simplesmente faz com que outras aplicações funcionem mal, sua remoção pode ser garantida. As duas plataformas permitem a remoção de patches indesejados, no entanto, os patches da Microsoft rastreiam os arquivos modificados, e desinstalar um patch exige apenas a utilização do recurso padrão de instalação/desinstalação do Windows e escolha por remover um patch. Infelizmente, essa capacidade não é uniformemente disponível para todos os softwares da Microsoft, mas está disponível apenas com os patches do Windows Server.

A RPM, ferramenta de gerenciamento de patch da Red Hat, também permite q remoção de patches; no entanto, é de responsabilidade do usuário especificar quais pacotes foram modificados pelo `up2date`.

A ferramenta `up2date` pode relacionar esses pacotes usando “--listas-rollbacks” na linha de comando.

A Oracle permite a reversão do patch através de sua ferramenta de linha de comando do `opatch`. Os usuários devem fornecer o número de ID de patch específico da plataforma, incluído no arquivo leiame do patch. Enquanto a ferramenta Oracle Enterprise Manager não suporta diretamente a reversão do patch, o processo é automatizado com execução remota de script por meio do Enterprise Manager.

Conclusões

Toda vez que implantamos uma nova tecnologia na empresa, pensamos em termos de complementar uma questão corporativa estratégica. Em lugar nenhum isso é mais verdadeiro do que no gerenciamento de dados, em que os bancos de dados relacionais transformam vastas quantidades de dados em informações que potencializam os negócios. Um dos principais desafios é que existem diversas tecnologias diferentes, plataformas e soluções que podem satisfazer uma série de requisitos corporativos, e toda solução tem um custo. Alguns desses custos, como o preço de compra, são fáceis de analisar, mas outros, como o custo da exposição a partir de falhas latentes nos sistemas, são muito mais complexos. Neste relatório, estudamos tanto os dados quantitativos como os qualitativos que afetam as vulnerabilidades e, assim, o risco de segurança operacional das diferentes plataformas de um servidor de banco de dados. Esses resultados fornecem certos aspectos adicionais da segurança do sistema a ser considerada ao se tomar decisões.

Mais especificamente, para a função do servidor de banco de dados, consideramos três configurações; Microsoft SQL Server 2000 no Windows Server 2003, Oracle 10g no Red Hat Enterprise Linux 3 e MySQL no Red Hat Enterprise Linux 3. Com o intuito de produzir uma comparação significativa entre as plataformas, os sistemas estudados foram manualmente instalados e suas configurações foram verificadas.

Ao considerar os dados quantitativos, examinamos o número e o tipo de vulnerabilidades reportadas em cada plataforma. Filtramos essas vulnerabilidades com base nos recursos e pacotes instalados em nossas três construções de sistema. Para cada vulnerabilidade, determinamos o tempo total decorrido entre a divulgação pública da vulnerabilidade e a disponibilidade do patch que solucionou a vulnerabilidade.

Em se falando de contagem de vulnerabilidades, o SQL Server 2000 na solução do Windows apresentou uma vantagem nítida sobre as soluções do Oracle e MySQL, construídas sobre o servidor Red Hat Linux. Enquanto uma deve considerar a segurança da plataforma subjacente para cada sistema de banco de dados implantado, torna-se ilustrativo checar as vulnerabilidades no software do próprio servidor de banco de dados. Esses dados apresentam uma vantagem pouco significativa ao SQL Server 2000 sobre o MySQL e Oracle.

No entanto, é interessante reparar que esses servidores de banco de dados estão em estágios diferentes de seus respectivos ciclos de vida. O SQL Server 2000 é utilizado desde 2000, ao passo que o Oracle obteve mais versões freqüentes, com seu servidor de banco de dados 10g sendo lançado no início de 2004. Uma estatística comprovada, portanto, é o *número total* de vulnerabilidades na vida do produto e *quando* esses problemas foram encontrados. O SQL Server 2000 teve um total de 36 vulnerabilidades reportadas em mais de 4 anos, a partir de seu lançamento, em 2000: 8 em 2000, 4 em 2001, 19 em 2002, 4 em 2004 e 1 em 2004. O Oracle totaliza 30 vulnerabilidades em seu primeiro ano de lançamento para seu servidor de banco de dados 10g, com 10 de suas vulnerabilidades sendo externamente atribuídas a um identificador CAN. No MySQL, os dados estão menos claros devido à natureza de suas versões incrementais. Analisando-se o período de um ano, terminando em 28 de Fevereiro de 2005, o servidor MySQL totalizou 7 vulnerabilidades distintas.

Além da contagem das vulnerabilidades, também analisamos o período médio e cumulativo de exposição às vulnerabilidades das três construções – a tão conhecida métrica dos Dias de Risco. Em se tratando de dias de risco médios, a Microsoft apresentou o menor resultado, de 32.0, seguida da Oracle no RHEL3, com 38.7 e a solução do

MySQL, com 61.6. Tanto o SQL Server 2000 no Windows Server 2003 como o Oracle 10g no RHEL 3 se beneficiaram dos dias de risco médios por meio de seu forte incentivo à "divulgação de responsabilidade", em que eles tentam, cuidadosamente, coordenar a publicação da vulnerabilidade, anunciando a solução e, ativamente, projetando relações com novos pesquisadores de segurança. Os dados do Red Hat mostram provas do levantamento da diretiva de divulgação de responsabilidade, com diversos dias de risco zero. Isso ajuda a controlar as médias de forma a reduzir diretamente o risco do cliente.

De forma qualitativa, descrevemos diversos fatores que, finalmente, conduzem a viabilidade de uma solução específica no que se refere à segurança. Enquanto esses aspectos de uma solução podem ser difíceis de ser considerados de maneira quantitativa, está claro que eles desempenham uma função de determinar o grau de facilidade para que a segurança possa ser gerenciada e mantida, surtindo impacto direto no risco geral. Cada organização precisa levar em consideração esses fatores qualitativos, assim como as métricas que podem ser atribuídas a um grande número, ao se tomar uma decisão de implantação.

Na comparação, como pesquisadores da segurança, sabemos que as soluções de banco de dados da Microsoft, Oracle e do mundo do código aberto, executado tanto na plataforma Red Hat como na Microsoft, podem ser usadas para fornecer uma solução segura quando implantada e administrada com os conhecimentos apropriados e sob as políticas corretas. Analisando os fatores de segurança do software que todo fornecedor tem a capacidade de afetar diretamente, no entanto, como a qualidade e resposta de segurança de um software, dos três sistemas de banco de dados estudados, o Microsoft SQL Server 2000 no Windows Server 2003 apresentou menos vulnerabilidades na segurança e menos dias de risco em comparação com as soluções do MySQL e Oracle no Red Hat Enterprise Linux 3.

É impossível (e irresponsável) fornecer uma comparação abrangente sobre as questões de segurança que serão aplicadas a esses ambientes operacionais. Nossa pesquisa mostrou que o perfil de ameaças que um pacote enfrenta pode ser um determinante fundamental ao se analisar a segurança em geral. Para tanto, a importância de cada fator contribuinte em relação à segurança deve ser pesada para cada ambiente de ameaça.

Tabela Combinada de Comparações

A tabela que segue resume as nossas descobertas a respeito das contagens de vulnerabilidades para as três configurações consideradas:

Severidade	Windows Server 2003/SQL Server 2000	RHEL ES 3/Oracle 10g	RHEL ES 3/MySQL 3.23.58
Alta	27	73	41
Média	18	63	49
Baixa	0	10	8
Desconhecida	18	61	18
Total	63	207	116

Tabela 3: Resumos da contagem de vulnerabilidades para as três soluções estudadas

A tabela abaixo resume os resultados de dias de risco para as três configurações consideradas:

Severidade	Windows Server 2003/SQL Server 2000	RHEL ES 3/Oracle 10g	RHEL ES 3/MySQL 3.23
Dias de Risco: Severidade Alta	952	2539	1525
Dias de Risco: Severidade Média	573	3314	3594
Dias de Risco: Severidade Baixa	0	574	741
Dias de Risco: Desconhecida	490	1590	1290
Dias de Risco Cumulativos	2015	8017	7150
Dias de Risco Médios Por Vulnerabilidade	31.98	38.73	61.64

Tabela 4: Resumo dos Dias de Risco para as três soluções estudadas

Apêndice A: Metodologia Passo a Passo

Nosso objetivo, nas comparações quantitativas de vulnerabilidades deste documento, foi de criar e seguir uma metodologia que estivesse clara para fornecer resultados significativos aos tomadores de decisão. Neste apêndice, incluímos um processo passo a passo para reconstruir os dados analisados neste relatório, assim como as métricas dos resultados.

Abaixo estão os passos que podem ser seguidos para projetar a sua própria série de dados para análise.

A. Elabore uma planilha de vulnerabilidades do Windows Server 2003

a. Seqüencialmente, examine cada Boletim de Segurança liberado pela Microsoft durante o período estudado, não contando com a lista de pesquisa do Boletim de Segurança fornecida. Os Boletins de Segurança Microsoft e as vulnerabilidades solucionadas por eles estão em: http://www.microsoft.com/technet/security/current.aspx.

b. Para cada Boletim, leia e identifique se o Windows Server 2003 é afetado. Geralmente, a Microsoft inclui uma tabela, no Sumário Executivo, do boletim, com uma linha para cada vulnerabilidade listada pelo Nome CVE e mostrando a severidade da Microsoft para cada plataforma afetada.

c. Para cada Nome CVE, preencha as seguintes colunas. Note que um Boletim que soluciona diversas vulnerabilidades resulta múltiplas linhas:

i. Nome CVE (ex. CAN-2004-1028)

ii. Identificador do Boletim de Segurança MSFT (ex. MS04-007)

iii. Data do Boletim de Segurança/Solução

d. Uma vez que supomos que todos os componentes estejam presentes no WS2003, não é necessário agrupar componentes em grupos de funções, como fazemos para o Red Hat.

e. Uma vez que supomos que todos os componentes estejam presentes no WS2003, não precisamos no passo de validação para verificar se o componente está fisicamente instalado. Presumimos que ele esteja instalado em todos os casos.

B. Elabore uma planilha de vulnerabilidades do Red Hat Enterprise Linux 3 Enterprise Server (RHEL3ES).

a. Seqüencialmente, examine cada pesquisa de segurança do RHEL3ES liberado pelo Red Hat durante o período estudado. As pesquisas de segurança do RHEL3ES e as vulnerabilidades solucionadas por eles estão em: https://rhn.redhat.com/errata/rhel3es-errata-security.html .

b. Para cada pesquisa de segurança, leia e confirme se RHEL3ES foi afetado. O identificador de pesquisa, o componente afetado e os Nomes CVE associados são geralmente listados no cabeçalho da pesquisa de segurança.

c. Para cada Nome CVE, Para cada Nome CVE, preencha as seguintes colunas. Note que uma pesquisa que soluciona diversas vulnerabilidades resulta múltiplas linhas:

i. Nome do CVE (ex. CAN-2004-1028)

ii. Identificador da Pesquisa de Segurança (ex. RHSA-2005:010)

iii. Data da Pesquisa de Segurança /Solução

iv. Cada pacote diagnosticado

C. Elabore uma planilha de vulnerabilidades do Oracle 10g

a. Seqüencialmente, examine cada pesquisa de segurança do Oracle para o 10g liberado pela Oracle durante o período estudado. As pesquisas de segurança do e as vulnerabilidades solucionadas por eles estão em: http://www.oracle.com/technology/deploy/security/alerts.htm .

b. Para cada pesquisa de segurança, leia e confirme se o Oracle 10g foi afetado. O identificador de vulnerabilidade é geralmente listado como DB*NN* em uma tabela próxima ao final do documento. Examine a coluna "Última Série de Patch Afetada" para determinar se a vulnerabilidade se aplica ao 10g.

c. Para cada vulnerabilidade, preencha as colunas que seguem. Note que uma pesquisa que soluciona diversas vulnerabilidades resulta múltiplas linhas:

i. Identificador da vulnerabilidade

ii. Identificador da Pesquisa de Segurança (ex. Alert 68)

iii. Data da Pesquisa de Segurança/Solução

d. Manualmente, procure por referências cruzadas aos identificadores CVE. OBSERVAÇÃO: este passo é difícil, sendo necessário apenas para os cálculos dos Dias de Risco; as contagens de vulnerabilidades podem ocorrer sem eles.

D. Reúna informações para Calcular os Dias de Risco

a. Para cada Nome CVE das planilhas do respectivo servidor, analise as referências listadas em http://cve.mitre.org. Por exemplo, os detalhes da CAN-2004-0021 estão relacionados em http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-0021.

b. Siga cada referência, examinando a data de publicação da página da Web referenciada que tornou a questão pública. Digite a data mais antiga na planilha como "Primeira data Pública" e a URL na planilha como "Primeira referência Pública".

c. Para as entradas dos bancos de dados com bugs, a entrada mais antiga não deve ser publicamente legível ao ser digitada. Primeiro verifique a política do servidor do bug para ver se as entradas estão sempre públicas. Se não, verifique uma entrada apresentando que a indicação de privacidade foi alterada para tornar o bug publicamente visível[19].

d. Uma vez que a lista do CVE não garante a captura da primeira referência pública, embora ela o faça, verifique as referências com qualquer referência listada

[19] Observe que este é um passo de nova metodologia adicionado à liberação deste relatório, para acomodar as vulnerabilidades no software do código aberto que foi mantido privado por meio do gerenciamento de bugs para depois se tornar público. Nossa metodologia considera a data pública como uma divulgação pública do bug.

Na pesquisa de segurança do Boletim. Se uma referência pública mais atual for encontrada, utilize-a, assim como a data mais atual.

e. Além disso, busque na Internet e nos grupos de notícias mais comuns, uma discussão pública da vulnerabilidade da segurança e utilize os dados e referências, caso encontrados.

f. Adicione uma coluna às planilhas chamadas "Dias de Risco" e subtraia a "Primeira Data Pública" da "Data do Boletim/solução" para calcular os dias de risco daquela vulnerabilidade.

E. Para a planilha do WS2003 e RHEL3ES, adicione a relação de severidade do ICAT.

a. Baixe o último ICAT Metabase de http://icat.nist.gov/icat.cfm.

b. Para cada Nome CVE, nas planilhas do servidor, digite um valor da coluna de ALTA, MÉDIA, BAIXA ou INDESEJADA

F. Para a planilha RHEL3ES, crie uma planilha para cada função do servidor.

a. Instale e elabore um sistema de RHEL3AS na configuração que está sendo estudada (ex. função do servidor da web). Use o comando 'rpm –qa' para verificar cada pacote que foi diagnosticado com alguma pesquisa de segurança durante o período.

Após concluir os cinco passos acima, você deve fazer as planilhas capturarem a lista de vulnerabilidades para cada função da plataforma e servidor, para certo período, juntamente com a classificação da severidade, primeira data de publicação, primeira referência pública e cálculo dos dias de risco. A partir disso, você pode calcular as contagens, os totais e as médias, se desejar.

Apêndice B

O texto que segue foi tirado dos procedimentos recomendados de instalação do Oracle 10g no Red Hat Enterprise Linux 3, podendo ser encontrado em http://www.oracle.com/technology/pub/articles/smiley_10gdb_install.html#rhel3. As informações abaixo são atuais desde 1.° de Maio de 2005.

RHEL3

O Oracle Database 10*g* é certificado para executar a versão base do Red Hat Enterprise Linux 3(Advanced Server e Enterprise Server) sem atualizações. Se você possui CDs de atualização, pode usar o CD de boot a partir da atualização, em vez de fazê-lo por meio da versão base, para aplicar automaticamente todas as atualizações durante a instalação. Todas as atualizações do Red Hat são suportadas pelo Oracle.

1. Reinicialize o servidor usando o primeiro CD.
   - Você pode precisar alterar as configurações de BIOS para permitir a reinicialização a partir do CD.
2. A tela de reinicialização aparece com o `boot:` prompt na parte superior.
   - Selecione **Enter** para continuar com a instalação gráfica no console. (Para os métodos e opções de instalação, recorra ao *Manual de Instalação do Red Hat*).
   - `O instalador pesquisa seu` hardware, exibe brevemente a tela do Red Hat e depois inicia uma série de prompts de tela.
3. Seleção do Idioma
   - Aceite o padrão.
4. Configuração do Teclado
   - Aceite o padrão.
5. Tela de Boas Vindas
   - Clique em **Next**.
6. Configuração do Mouse
   - Aceite o padrão.
7. Tipo de Instalação
   - Selecione **Custom**.
8. Configuração de Particionamento de Disco
   - Um tratamento completo da partição de disco está além do escopo deste manual, que supõe que você seja familiarizado com os métodos de particionamento de disco.

     (CUIDADO: Particionar incorretamente um disco é uma das maneiras mais claras e rápidas de **destruir tudo o que estiver em seu disco rígido**. Caso você não tenha certeza de como proceder, pare e peça ajuda, ou irá se arriscar a perder seus dados!)

     Este manual utiliza o seguinte esquema de particionamento, com o ext3 para cada filesystem:

     O disco de 9GB no primeiro controlador (/dev/sda) irá suportar todos os softwares do Linux e
     Oracle, contendo as seguintes partições:
     - 100MB /partição de boot
     -1,500MB partição de troca—Defina-a a, no mínimo, duas vezes da quantidade de RAM

No sistema, mas não a mais de 2GB (os sistemas de 32 bits não suportam arquivos de troca com mais de 2GB). Caso você precise de mais 2GB de espaço de troca, crie múltiplas partições de troca.
-7,150MB de partição raiz — Essa partição será usada para tudo, incluindo /usr, /tmp, /var, /opt, /home, e mais. Isso foi feito apenas para simplificar a instalação para fins deste manual. Um esquema de particionamento mais robusto iria separar esses diretórios em filesystems isolados.

9. Configuração Mais Carregada de Boot
   - Aceite o padrão
10. Configuração da Rede
    - É geralmente melhor configurar os servidores de banco de dados com um endereço IP estático. Para isso, clique em **Edit**.
    - Uma janela de pop-up aparece. Desmarque a caixa **Configure using DHCP**, e digite um endereço IP e uma Máscara de Rede para o servidor. Veja se a opção **Activate on boot** está selecionada, e clique em **OK**.
    - Na caixa Hostname, selecione **manually** e digite o hostname.
    - Na caixa Miscellaneous Settings, digite as configurações restantes de rede.
11. Configuração do Firewall
    - Para fins de análise, nenhum firewall será configurado. Selecione **No firewall**.
12. Suporte Adicional a Idioma
    - Aceite o padrão.
13. Seleção de Fuso Horário
    - Escolha as configurações de hora mais apropriadas para a sua região. Configurar o sistema para UTC é geralmente uma boa prática para os servidores. Para isso, clique em **System clock uses UTC**.
14. Configure a Senha Raiz
    - Digite uma senha da raiz, e digite-a novamente para confirma-la.
15. Seleção do Grupo do Pacote
    - Selecione apenas as séries de pacotes mostradas aqui. Mantenha as outras sem seleção.
    - Desktop
      - X Window System
      - Gnome
      - KDE
      - Veja meus comentários na seção RHES 2.1, independente da escolha da GUI.
    - Aplicativos
      - Editores
      - Internet Gráfica
    - Servidores
      - Não selecione nada neste grupo.
    - Desenvolvimento
      - Ferramentas de Desenvolvimento
    - Sistema
      - Ferramentas Administrativas
    - Red Hat Enterprise Linux
      - Não selecione nada neste grupo.
    - Miscelânea
      - Desenvolvimento do Software de Legado
    - Clique em **Next** para prosseguir.
16. Sobre a Instalação
    - Clique em **Next**.
17. Instalando Pacotes

- O software será copiado no disco rígido e instalado. Mude os discos quando for notificado e clique em **Next** quando a instalação estiver concluída.

18. Configuração da Interface Gráfica (X)
    - Aceite os padrões, a menos que o instalador não reconheça a sua placa de vídeo. Caso isso aconteça, você não conseguirá continuar.
19. Configuração do Monitor
    - Aceite o padrão se o instalador identificar corretamente o seu monitor. Caso contrário selecione um monitor compatível a partir da lista.
20. Personalize a Configuração Gráfica
    - Aceite os padrões.
21. Congratulações
    - Remova a mídia de instalação do sistema e clique em **Next**.
22. O sistema reinicia automaticamente e apresenta a nova tela de boas vindas.
    - Clique em **Next**.
23. Acordo de Licença
    - Leia o acordo de licença. Se concordar com os termos, selecione **Yes, I agree to the License Agreement** e clique em **Next**.
24. Data e Hora
    - Defina a Data e a Hora.
    - Caso queira usar um servidor NTP (recomendado), selecione **Enable Network Time Protocol** e digite o nome do servidor NTP.
25. Conta de Usuário
    - Crie uma conta para você.
      - Não crie uma conta para o Oracle agora. Isso será mostrado mais adiante nesta seção.
26. Rede do Red Hat
    - Caso queira usar ou ativar a conta da sua Rede do Red Hat agora, aceite o padrão, clique em **Next**, e siga as instruções de ativação do produto que acompanham o Red Hat.
27. CDs adicionais
    - Clique em **Next**.
28. Conclua a configuração
    - Clique em **Next**.
29. Aparecerá uma tela gráfica de login.
30. Parabéns! Seu software do Linux está instalado.

Verificando Sua Instalação

Se você concluiu os passos acima, deve ter todos os pacotes e atualizações exigidos pelo Oracle Database 10*g*. No entanto, pode executar alguns passos para verificar a sua instalação.

Versão kernel requerida: 2.4.21-4.EL (É a versão kernel que vem com a liberação base do RHEL3. Este kernel, ou qualquer outro fornecido nas atualizações, funciona com o Oracle Database 10*g*.)

Verifique a versão do seu kernel, executando o seguinte comando:

```
uname -r

Ex:
# uname -r
2.4.21-4.0.1.ELsmp
```

Outras versões exigidas do pacote (ou superiores):

gcc-3.2.3-2
make-3.79
binutils-2.11
openmotif-2.2.2-16
setarch-1.3-1
compat-gcc-7.3-2.96.122
compat-gcc-c++-7.3-2.96.122
compat-libstdc++-7.3-2.96.122
compat-libstdc++-devel-7.3-2.96.122
compat-db-4.0.14.5 (listado no *Manual de Instalação do Oracle 10g Database* como exigido, mas não é necessário aqui)

Para ver as versões que estão instaladas desses pacotes em seu sistema, execute os seguintes comandos como raízes:

```
rpm -q gcc make binutils openmotif setarch compat-db compat-gcc \
     compat-gcc-c++ compat-libstdc++ compat-libstdc++-devel

Ex:
# rpm -q gcc make binutils openmotif setarch compat-db compat-gcc
\
>      openmotif  compat-gcc-c++  compat-libstdc++  compat-
libstdc++-devel
gcc-3.2.3-20
make-3.79.1-17
binutils-2.14.90.0.4-26
openmotif-2.2.2-16
setarch-1.3-1
package compat-db is not installed
compat-gcc-7.3-2.96.122
compat-gcc-c++-7.3-2.96.122
compat-libstdc++-7.3-2.96.122
compat-libstdc++-devel-7.3-2.96.122
```

Observe que o pacote compat-db não está instalado. Este pacote não está disponível em nenhum dos grupos disponíveis durante a instalação e deve ser instalado separadamente. Se alguma outra versão do pacote estiver faltando em seu sistema, ou as versões são mais antigas do que as especificadas acima, (que não seja o compat-db), você pode baixar e instalar as atualizações da Rede do Red Hat.

Instalando o compat-db

Insira o CD2 da mídia original do Red Hat Enterprise Linux. (Este pacote não foi atualizado desde a Atualização 2, sendo encontrado apenas na mídia original).

O CD é executado automaticamente.

Execute o seguinte comando como raiz:

```
rpm -ivh /mnt/cdrom/RedHat/RPMS/compat-db-4.0.14-5.i386.rpm

Ex:
# rpm -ivh /mnt/cdrom/RedHat/RPMS/compat-db-4.0.14-5.i386.rpm
```

```
Preparing...                ############################################
[100%]
   1:compat-db              ############################################
[100%]
```